



Programação Tangível no Mundo Físico: TaPrEC + Sphero

Title: Tangible Programming in the Physical Environment: TaPrEC + Sphero

Bruna Z. Panaggio
Instituto de Computação,
Universidade Estadual de
Campinas
brunazp@gmail.com

Marleny Luque Carbajal
Instituto de Computação,
Universidade Estadual de
Campinas
marleny.carbajal@
students.ic.unicamp.br

M. Cecília C. Baranauskas
Instituto de Computação,
Universidade Estadual de
Campinas
cecilia@ic.unicamp.br

Resumo

Apesar de acostumados a consumir mídias digitais e interagir com novas tecnologias, poucos estudantes são desafiados a criar seus próprios projetos tecnológicos. O Pensamento Computacional é um conceito e ferramenta que pode auxiliar o aprendizado de importantes estratégias de planejamento e de resolução de problemas, incluindo soluções tecnológicas. Uma maneira de introduzir as crianças ao pensamento computacional é por meio de ambientes de programação tangível. Tais ambientes em geral exploram a tangibilidade na entrada, isto é, na construção do programa por meio de Interfaces Tangíveis. Neste trabalho, buscamos explorar a tangibilidade do ambiente de programação tangível TaPrEC estendendo-a para a fase de execução, com feedback no mundo físico. Apresentamos a integração do TaPrEC com o Sphero, uma bola robótica que permite conexão com outros dispositivos. Para avaliar este novo sistema, realizamos 6 oficinas com professoras e crianças de um espaço educativo. Os resultados das oficinas são encorajadores e apontam para novas possibilidades de ambientes tangíveis que expandam habilidades do pensamento computacional.

Palavras-Chave: Programação Tangível; Pensamento Computacional; Educação; Interfaces Tangíveis; Sphero.

Abstract

Despite being familiar with consuming digital media and interacting with new technology, only a minority of students are challenged to create their own technological projects. Computational Thinking is a concept and tool that can help the students to learn some important strategies regarding planning and problem-solving, including technological solutions. One way to introduce young children to the Computational Thinking is through tangible programming environments. Usually, these environments exploit the tangibility at the entrance, i.e., in the construction phase of the program, through Tangible Interfaces. In this paper, we seek to explore the tangibility of the TaPrEC, a tangible programming environment, by extending it to the execution phase, providing feedback in the physical world. We present the integration of TaPrEC with Sphero, a robotic ball that connects to other devices. To evaluate this new system, we conducted 6 workshops with teachers and children in a learning space. The workshops results are encouraging and suggest new possibilities for tangible environments that expand computational thinking skills.

Keywords: Tangible Programming; Computational Thinking; Education; Tangible User Interface; Sphero.



1 Introdução

Os estudantes de hoje fazem parte de uma geração que já tem crescido com as novas tecnologias, usando computadores, videogames, tocadores de música digital, câmeras de vídeo, smartphones e outros tipos de brinquedos e ferramentas da era digital. Por essa fluência com as tecnologias digitais, são conhecidos como “nativos digitais” (Prensky, 2001). Entretanto, como apontado por Resnick et al. (2009), apesar de jovens e crianças estarem muito confortáveis em consumir mídias digitais e interagir com novas tecnologias, poucos são capazes de criar e projetar suas próprias aplicações tecnológicas. Resnick et al. (2009) afirmam que a programação tem o potencial de permitir que as pessoas possam ser criadoras e não apenas consumidores de tecnologia, favorecendo resultados positivos para o aluno. Outros estudos mostraram que aprender a programar tem um impacto positivo na criatividade e no desenvolvimento de habilidades cognitivas e socioemocionais (Liao e Bright, 1991).

Papert (1980) reconheceu que a programação de computadores como uma atividade educacional tinha um grande potencial como veículo para aquisição de habilidades cognitivas tais como a resolução de problemas e a atitude reflexiva. Wing (2014) definiu pensamento computacional como o “processo de pensamento envolvido na formulação de um problema e na expressão de sua solução (ou soluções) de tal forma que um computador (humano ou máquina) possa executar de forma eficaz” (tradução dos autores). Como defendido por Wing (2006), o pensamento computacional é uma ferramenta fundamental para todos, não somente para cientistas da computação, uma vez que possibilita aprender importantes estratégias de planejamento e de resolução de problemas fazendo uso dos conceitos fundamentais da Ciência da Computação (como abstração, decomposição, modularização e design iterativo, por exemplo), que servem para domínios que não estão somente relacionados à programação. Valente et al. (2017) investigaram o contexto geopolítico e as questões econômicas, que estão envolvidos no cenário atual da presença do pensamento educacional na educação. Dada a importância do pensamento computacional, criar ambientes que tornem seu aprendizado facilitado, mais agradável e mais eficiente e que possibilitem introduzi-lo nas salas de aula torna-se muito relevante à educação contemporânea.

Uma maneira potencialmente facilitada de introduzir as crianças à programação e ao pensamento computacional são as Interfaces Tangíveis. As Interfaces Tangíveis (TUIs – *Tangible User Interfaces*) permitem que as pessoas utilizem objetos físicos e do ambiente para interagir com informações digitais, ao invés de manipular periféricos tradicionais como mouse, teclado e monitor para acessá-las (Ishii e Ullmer, 1997). Em cenários de uso de tecnologias contemporâneas, as Interfaces Tangíveis dão forma física à informação digital ao empregar artefatos físicos que servem como representações e controles das correspondentes digitais. As TUIs tornam a informação digital diretamente manipulável e perceptível por meio dos outros sentidos (Ishii, 2008). As Interfaces Tangíveis são utilizadas em aplicações variadas, citando apenas alguns: como em um sistema que permite efetuar avaliação de estados afetivos de maneira lúdica utilizando bonecos tridimensionais (Moreira et al., 2019), em um ambiente de narrativa tangível proposto para estruturar narrativas com o uso de elementos de interatividade (Baranauskas e Posada, 2017), para auxiliar o aprendizado de leitura para crianças (Fan et al., 2016) e para aprimorar a experiência de crianças no aprendizado de idiomas mediante o uso de atividades que envolvem leitura e vocabulário (Sánchez et al., 2011).

As Interfaces Tangíveis são capazes de promover um engajamento mais forte e de longa duração com um maior potencial para envolver as crianças durante atividades de aprendizado (Sylla et al., 2012), além de envolver todos os sentidos, apoiando o desenvolvimento geral da criança (Shaer e Hornecker, 2009) e evocar sentimentos de felicidade, alegria e motivação (Posada et al., 2014). O aumento da capacidade de processamento dos dispositivos embarcados, aliado à



miniaturização e barateamento desses recursos têm levado ao surgimento de tecnologias contemporâneas, de modo a deixar os artefatos tangíveis mais acessíveis.

O TaPrEC, acrônimo de *Tangible Programming Environment for Children*, (Carbajal e Baranauskas, 2015) é um ambiente de programação tangível de baixo custo desenvolvido para explorar o pensamento algorítmico e para ensinar conceitos básicos de programação para crianças. O ambiente foi avaliado com crianças e professores de ensino fundamental em contexto educacional real. Os usuários podem criar programas de computador tangíveis organizando blocos de madeira semelhantes a peças de quebra-cabeça, contendo tecnologia de RFID (*Radio Frequency Identification*). No TaPrEC, o resultado da execução do programa tangível se mostra de maneira virtual, e é apresentado no contexto de programação *Scratch*¹.

Segundo Zuckerman *et al.* (2005), algumas vantagens trazidas pelas TUIs para a educação são: i) o engajamento sensorial, as crianças aprendem de forma natural, usando vários sentidos (toque, visão, audição) em um processo construtivo que aumenta a retenção de conteúdo; ii) a acessibilidade, já que as TUIs fornecem mais opções a crianças com necessidades especiais; iii) a aprendizagem em grupo, TUIs facilitam trabalho colaborativo e discussões. Além disso, teóricos da educação, como Papert (1980), afirmam que as atividades que envolvem a robótica têm um tremendo potencial para melhorar o ensino em sala de aula. Os autores Rusk *et al.* (2008) concordam que a forma como a robótica é introduzida em ambientes educacionais é desnecessariamente estreita. Portanto, explorar uma gama mais ampla de possíveis aplicações tem o potencial de envolver os alunos e gerar um maior interesse: os alunos tornam-se motivados quando as atividades robóticas são introduzidas de maneiras diferentes como por exemplo para contar uma história ou em conexão com outras disciplinas e áreas de interesse, como música e arte. Em vista disso, argumentamos que a consideração da tangibilidade do ambiente TaPrEC tanto na sua entrada como na sua saída, poderia melhorar significativamente a experiência do usuário assim como o engajamento sensorial, a aprendizagem em grupo e a acessibilidade.

Em um estudo exploratório, Panaggio e Baranauskas (2017) estudaram as possibilidades de uso de uma bola robótica, denominada Sphero², em contexto escolar, visando entender se as atividades que utilizassem tal dispositivo podiam ser divertidas e motivadoras para crianças. Considerando os resultados positivos de tal pesquisa, bem como entendendo a importância do pensamento computacional e os benefícios dos sistemas tangíveis, buscamos neste trabalho investigar a tangibilidade na execução dos comandos no mundo físico e o potencial de engajamento dos participantes na experiência de interação no ambiente. Para tal expandimos o ambiente de programação tangível TaPrEC para promover a tangibilidade também na execução dos programas por meio da bola robótica Sphero. O método de pesquisa envolveu um processo iterativo de design e desenvolvimento incremental do ambiente, seguido de um estudo exploratório de uso do ambiente em contexto educacional. Resultados do estudo, obtidos a partir da resposta afetiva dos participantes em Oficinas, da observação de pesquisadores *in loco* e análise de vídeos, mostraram a efetividade do ambiente proposto e sugerem novas oportunidades de pesquisa na área.

Assim, este artigo está organizado do seguinte modo: Na seção 2 a seguir, apresentamos trabalhos relacionados que ajudam a colocar luz a esta pesquisa. Em seguida, na seção 3 descrevemos o design e desenvolvimento do sistema utilizado neste trabalho. Na sequência, na seção 4 apresentamos o estudo exploratório realizado para investigar o sistema desenvolvido em uso. Na seção 5 sintetizamos os resultados e discutimos implicações deste trabalho. Por fim, na seção 6 apresentamos as conclusões.

¹ Scratch. <https://scratch.mit.edu/>. Último acesso em maio de 2018.

² Sphero. <https://www.sphero.com/>. Último acesso em maio de 2018.



2 Programação, Blocos e Tecnologias

O termo “Programação Tangível” foi usado pela primeira vez por Suzuki e Kato, que desenvolveram o *AlgoBlock* (Suzuki e Kato, 1995), uma linguagem de programação tangível cujo foco é promover a resolução colaborativa de problemas, facilitando a interação entre os estudantes. Para criar um programa, as crianças conectam blocos de alumínio, sendo que para cada bloco é atribuído um comando de uma linguagem de programação muito semelhante ao Logo (Papert, 1980). Além dos blocos, o ambiente de programação conta com controlador, que conecta os blocos de programação a um computador e um monitor. Quando um botão para começar o programa é pressionado, é apresentado no monitor um submarino, que se desloca em um labirinto debaixo da água, de acordo com os comandos dos blocos. Apesar da programação ser feita de modo tangível, a execução do programa é mostrada em ambiente virtual.

O *T-Maze* (Wang et al., 2014) é uma ferramenta de programação tangível desenvolvida para crianças entre 5 e 6 anos para ajudá-las a explorar o pensamento computacional. O sistema permite que as crianças brinquem de escapar de um labirinto virtual utilizando blocos de programação de madeira, que contém um símbolo que identifica a funcionalidade do bloco, um texto simples e um identificador para o sistema de visão computacional. Os blocos também contam com pequenos ímãs nas laterais, de maneira que a atração e repulsão magnética auxiliam a criança a perceber que os blocos estão posicionados de maneira correta. Uma câmera captura os blocos de programação montados, e um sistema de visão computacional é utilizado para converter o programa físico em um programa virtual de forma automática. O sistema também permite que as crianças utilizem os blocos de madeira para criar seus próprios labirintos.

O *TanProRobot* (Wang et al., 2015) é um sistema tangível desenvolvido para ensinar conceitos de programação para crianças. O sistema é composto por três partes: os blocos de programação, um robô em formato de carro e itens de manipulação. Desse modo, a entrada e saída do sistema são tangíveis. As crianças podem programar o robô para fazer certas ações ao arrumar os blocos de programação de determinada maneira. O sistema também permite que as crianças interajam com o robô com o uso dos itens de manipulação. Cada bloco de programação possui um microcontrolador, que conecta um transmissor e um receptor infravermelho e um módulo sem fio. O infravermelho é utilizado para a comunicação entre os blocos: um bloco transmite informações por meio de sinal infravermelho para o bloco seguinte, que será ativado. Em seguida, os blocos transmitem os comandos através da rede sem fio. Quando o robô está em modo de programação, esses comandos são recebidos e armazenados pela unidade de processamento do robô. Quando o modo de execução é ativado, o robô passa a executar tais comandos.

O sistema *Robo-Blocks* (Sipitakiat e Nusen, 2012) permite que crianças controlem o movimento de um robô, ao juntar blocos físicos que representam ações. Os movimentos básicos do robô são: mover-se para frente, mover-se para trás, virar à direita e virar à esquerda. Os usuários podem determinar o quanto o robô irá andar, ou o quanto ele vai girar ao ajustar o número em um visor da peça movimentando um botão. Os blocos de ações se encaixam por conexões magnéticas, e a cadeia de blocos é anexada a um bloco mestre, que interpreta o programa e transmite os comandos para o robô.

O projeto *Spherly* (Trower e Gray, 2015) é um ambiente de programação visual para construir programas com o uso de blocos virtuais. Muito semelhante ao Scratch, a linguagem de programação proposta no Spherly possui blocos virtuais que controlam o Sphero e podem ser organizados sequencialmente na tela do computador. O ambiente é composto por dois elementos: uma aplicação web, na qual é possível manipular os blocos virtuais, que é iniciada por um navegador de Internet, e um servidor que é executado no computador, e que se comunica com o Sphero através do Bluetooth.



Já o *CodeSnaps* (Sabourin et al., 2016) é um ambiente de programação tangível, no qual os usuários podem escrever programas para movimentar o Sphero com blocos de papel. Cada bloco de papel possui um código óptico que identificava o comando daquele bloco. Os comandos são específicos para o Sphero: para fazer a bola robótica se locomover, é preciso indicar a direção, a velocidade e o tempo que a bola deve ficar percorrendo. Depois que o programa é construído com os blocos de papel, ele deve ser escaneado utilizando a câmera de um dispositivo móvel. Um programa digital, com blocos virtuais, é exibido no aplicativo, e o usuário deve interagir com o smartphone para rodar o programa. O aplicativo se comunica com o Sphero através de uma conexão Bluetooth, e a bola robótica executa os comandos do programa.

Embora uma revisão sistemática de literatura esteja fora do escopo deste trabalho, é possível verificar a presença de vários sistemas de programação tangível, com diferentes características em termos de sua tecnologia e modo de possibilitar a interação. A Tabela 1 sumariza características dos ambientes de programação encontrados organizadas em duas dimensões: 1) quanto à tangibilidade na entrada, na saída ou em ambos; 2) quanto ao nível de complexidade e abstração dos comandos. Consideramos nessa dimensão a complexidade das operações envolvidas, por exemplo, comandos como rotação do elemento programado ou alterações de velocidade e tempo para realização dos movimentos do item programado apresentam alta complexidade de abstração.

Tabela 1: Características de trabalhos relacionados.

Ambiente	Etapas de programação e execução tangíveis	(baixa) Complexidade e (alta) Abstração dos Comandos de programação
AlgoBlock (Suzuki e Kato, 1995)		✓
T-Maze (Wang et al., 2014)		✓
TanProRobot (Wang et al., 2015)	✓	
Robo-Blocks (Sipitakiat e Nusen, 2012)	✓	
Spherly (Trower e Gray, 2015)		
CodeSnaps (Sabourin et al., 2016)	✓	
TaPrEC (Carbajal e Baranauskas, 2015)		✓

Alguns dos sistemas promovem a tangibilidade apenas na etapa de programação, mantendo a execução de forma digital (como o *AlgoBlocks* e o *T-Maze*), enquanto outros permitem apenas a execução no mundo físico (como o *Spherly*). Já os sistemas que são tangíveis nas duas fases possibilitam que os usuários programem utilizando conceitos de direção (ou seja, envolvendo operação de rotação do elemento), velocidade e tempo (como o *TanProRobot*, o *RoboBlocks* e o *CodeSnaps*). Além disso, alguns dos sistemas possuem tecnologia embutida nos blocos de programação, o que os poderia deixar mais caros. O sistema proposto nesse trabalho apresenta tangibilidade nas duas etapas (na programação e no feedback da execução) e, por utilizar os comandos do TaPrEC como base para a programação, apresenta um nível de abstração mais alto e consequentemente baixa complexidade nas operações.



3 Design e Desenvolvimento do TaPrEC + Sphero

Nas subseções a seguir apresentamos o processo de design e desenvolvimento do sistema TaPrEC + Sphero. Iniciamos apresentando uma visão geral do ambiente de programação tangível TaPrEC; em seguida detalhamos o funcionamento do Sphero e finalizamos detalhando a integração entre o TaPrEC e Sphero. O processo seguiu metodologia incremental de design de sistemas, baseada em prototipagens iterativas alimentadas por testes-piloto.

3.1 TaPrEC

O TaPrEC foi proposto com o objetivo de facilitar a crianças o aprendizado de conceitos básicos de programação, sendo uma alternativa de baixo custo para o ensino de programação em escolas e potencialmente permitindo uma transição mais suave do ambiente tangível para ambientes de aprendizado virtuais e o mundo da programação (Baranauskas e Carbajal, 2017).

O ambiente de programação permite que crianças criem programas de computador ao organizar objetos tangíveis, trabalhando 3 conceitos básicos de programação: sequências, repetições e procedimentos (Carbajal e Baranauskas, 2015). O TaPrEC possui três elementos principais (Figura 1) que compõem sua arquitetura: um hardware de baixo custo, os blocos de programação com tecnologia embutida e um software de controle.

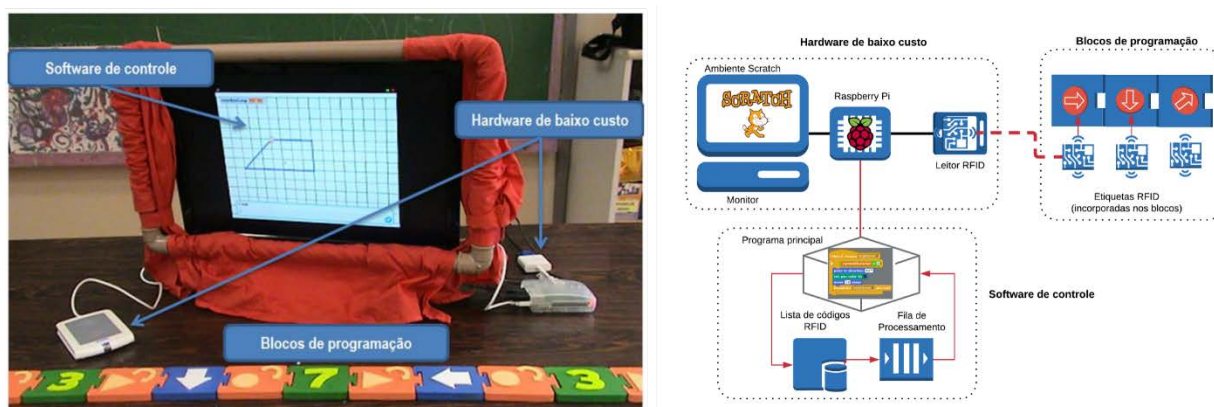


Figura 1: À esquerda, o ambiente físico do TaPrEC; à direita, a arquitetura do sistema. Fonte: Baranauskas e Carbajal, 2017.

O hardware do TaPrEC consiste em uma Raspberry Pi 2B Model³, computador de placa única de baixo custo, e de um leitor de RFID. O funcionamento de sistemas RFID é simples: uma etiqueta RFID, que contém um identificador, gera um sinal de radiofrequência com essa informação; este sinal é detectado pelo leitor de RFID, que envia a informação, agora em formato digital, para uma aplicação de software.

Os blocos de programação são um conjunto de blocos de madeira coloridos no formato de peças de quebra cabeças que contém um símbolo gráfico em alto relevo em uma face e uma etiqueta RFID no verso. Os blocos de programação estão agrupados de acordo a sua funcionalidade e com uma cor característica: cinza para os blocos de “controle”, azul para os blocos de “deslocamento”, laranja para os blocos de “repetições”, verde para os blocos de “números” e amarelo para os blocos de “procedimentos” (Figura 2). Para criar um programa, é necessário colocar as peças de madeira numa sequência específica: primeiro o bloco de ‘início’, depois os blocos de ações e finalmente o bloco de ‘fim’.

³ Raspberry Pi. <https://www.raspberrypi.com>. Último acesso em maio de 2018.

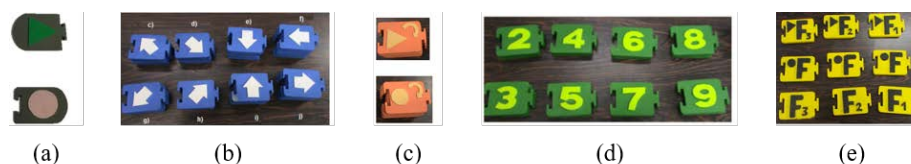


Figura 2: Blocos de programação do ambiente TaPrEC: (a) Controle, (b) Deslocamento, (c) Repetição, (d) Números, (e) Procedimentos. Fonte: Carbajal e Baranauskas, 2015.

Já o software de controle foi desenvolvido na linguagem de programação Scratch 1.4 (Resnick et al., 2009) e é responsável por receber a informação de qual bloco de programação foi lido com o leitor de RFID e armazená-la em uma lista. Quando o bloco de ‘início’ é lido, a lista é criada. Em seguida, quando os blocos de ação são lidos, seus identificadores são lidos e armazenados na lista. Por fim, no momento em que o bloco de ‘fim’ é lido, a lista é processada, verificando se o identificador está armazenado no sistema, e a qual ação ele corresponde. Em seguida, o software de controle executa o código referente a essa ação processada. A execução do programa se dá de maneira virtual, no contexto de programação Scratch, com saída projetada em um monitor ou televisão conectado ao sistema.

3.2 Sphero

O Sphero (Figura 3) é um ‘brinquedo inteligente’, no formato de uma bola robótica que se conecta através de conexão Bluetooth a outros dispositivos compatíveis, como smartphones e tablets. Quando conectado, é possível controlá-lo, alterando sua cor e fazendo o dispositivo se movimentar em várias direções e em diferentes velocidades. Além disso, o Sphero possui alguns sensores embutidos, como giroscópio e acelerômetro, que possibilitam algumas outras funcionalidades, como percepção da manipulação, detecção de colisão e queda, podendo ser utilizado como um controle ou para monitorar movimentos realizados com a mão. Por ser um objeto manipulável e controlável, o Sphero também pode ser considerado uma TUI.

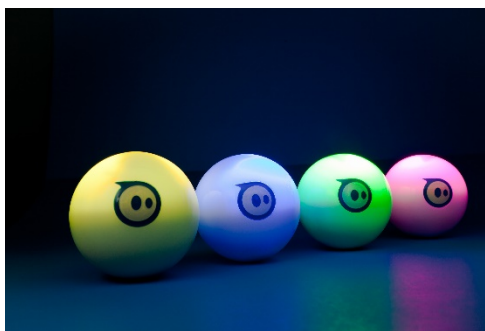


Figura 3: Spheros. Fonte: <http://bit.ly/2JdVLOj>. Acesso em maio de 2019.

Existem aplicativos para smartphones disponíveis no mercado e que utilizam o Sphero. O aplicativo *Sphero Play*⁴, permite que a bola robótica seja controlada e também que sirva como um controle. Quando a bola é controlada, é possível alterar sua cor, bem como realizar manobras com ela, alterando a direção, velocidade e duração dos movimentos realizados. Já quando o dispositivo é utilizado como um controle, os movimentos realizados com a bola são mapeados para movimentos de uma nave ou carro dentro do aplicativo. Um outro exemplo de aplicativo disponível no mercado é o aplicativo *Sphero Edu – Coding for Sphero Robots*⁵, que possibilita a

⁴ Sphero Play. <https://play.google.com/store/apps/details?id=com.sphero.spheromini>. Último acesso em maio de 2019.

⁵ Sphero Edu: <https://play.google.com/store/apps/details?id=com.sphero.sprk>. Último acesso em maio de 2019.



criação de programas com o uso de uma linguagem de programação baseada em blocos virtuais, para movimentar o Sphero, controlando a duração, a velocidade e direção dos movimentos.

Apesar da execução acontecer no mundo físico, esse aplicativo oferece um modo de programação digital, já que o usuário deve concatenar blocos de programação virtuais na tela do aplicativo. Além disso, é preciso conhecer algumas operações da linguagem, como definir a rotação que a bola deve fazer, além de definir o tempo e a velocidade do movimento. Com a integração do TaPrEC e do Sphero, o sistema possui programação e execução no mundo físico, ocultando tecnologias mais tradicionais (como computador e smartphone) das crianças, além de promover maior abstração de operações complexas da programação do robô, utilizando comandos mais simples, focando assim no aprendizado de habilidades básicas do pensamento computacional, abstraindo a complexidade de determinadas operações.

3.3 Integrando o TaPrEC e o Sphero

Para evoluir o TaPrEC e realizar a integração com o Sphero (Figura 4), foi necessária uma atualização do hardware, uma vez que a Raspberry Pi 2B Model utilizada no projeto original não possuía conexão Bluetooth. Para isso, foi necessário utilizar uma Raspberry Pi 3, que já possui conexão Bluetooth integrada.

Em seguida, foi preciso atualizar o software de controle para, ao invés de exibir as ações correspondentes aos blocos de programação no ambiente Scratch, fazer o Sphero se movimentar segundo tais ações (Figura 4, à direita). Para tal, foi utilizada uma biblioteca em Ruby, que age como uma interface entre o Sphero e a linguagem Scratch. Quando o Scratch reconhece a ação de um bloco de programação, ele envia um comando para essa biblioteca, que é responsável por enviar comandos que o Sphero entenda. Por exemplo, se um bloco de programação indica a ação “movimento para a direita”, o Scratch enviará um comando “move_right” para a biblioteca Ruby, que enviará para o Sphero um comando para a bola robótica girar 90° e um comando para se mover.

Portanto, para o funcionamento correto do sistema é necessário ligar e conectar o Sphero e a Raspberry Pi através de uma conexão Bluetooth. Em seguida é necessário colocar em funcionamento o ambiente TaPrEC e a biblioteca Ruby. Por fim, é necessário ajustar, manualmente, o ângulo inicial do Sphero. Depois desses passos serem executados, basta criar um programa com os blocos de programação de madeira, realizar a leitura deles com o leitor de RFID e, quando o bloco de ‘fim’ é lido, o Sphero começa automaticamente a executar os comandos.

Também foi necessário o desenvolvimento de um “tabuleiro”, para simular o ambiente virtual que as crianças experimentaram ao utilizar o TaPrEC: as crianças planejam as soluções para tarefas baseando-se em quantos quadrados devem ser avançados. Além de facilitar o planejamento, o tabuleiro se torna importante para deixar o solo mais estável, diminuindo a interferência do tipo de solo no movimento do Sphero.

Para movimentar o Sphero, é necessário fornecer à biblioteca Ruby com qual velocidade a bola robótica irá se locomover e por quanto tempo, ao invés de definir a distância a ser percorrida. Para tal, foram realizados alguns testes para definir os valores desses parâmetros, e permitir que o Sphero fizesse o percurso do centro de uma casa no tabuleiro para o centro de qualquer casa adjacente. Entretanto, dependendo da superfície em que o Sphero é utilizado, a resposta do robô pode variar. Por exemplo, se a superfície tiver alguma inclinação, o Sphero pode percorrer uma distância maior ou menor do que realmente deveria. Por seus comandos serem baseados em velocidade e tempo, uma superfície inclinada poderia fazer com que a bola ganhasse ou perdesse velocidade em relação ao que havia sido pré-definido. Como o tempo continua fixo,



a distância percorrida será maior ou menor do que a inicialmente planejada quando os parâmetros de velocidade e tempo foram definidos.

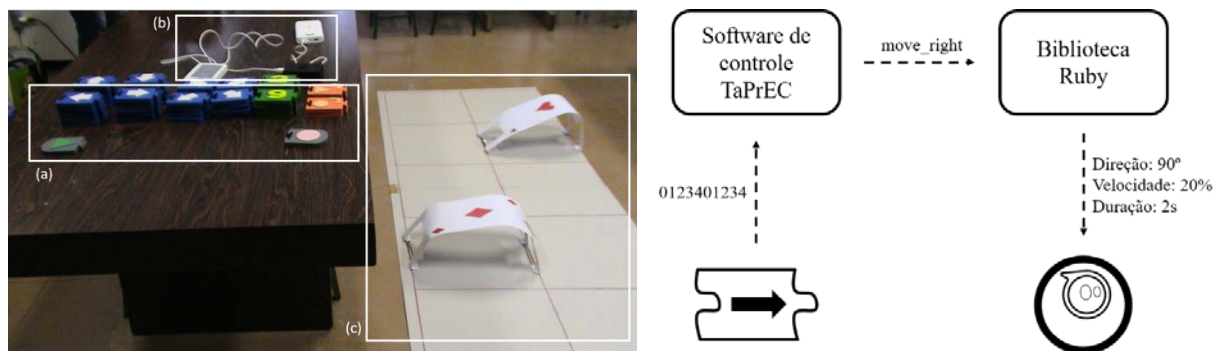


Figura 4: À esquerda, alguns componentes do sistema: (a) blocos de programação; (b) leitor RFID, Raspberry Pi 3 e fonte de energia; (c) tabuleiro. À direita, a arquitetura da integração dos sistemas. Fonte: dos autores.

3.3.1 Primeiro Protótipo: Efeitos do Tabuleiro no Sistema

O primeiro protótipo do tabuleiro, de 1,20 m x 1,20 m (Figura 5) foi construído com 6 folhas de 60 cm x 40 cm de EVA. Este material foi escolhido por ser acessível e de fácil manuseio. Para definir os limites de cada uma das casas e para unir as folhas foi utilizada fita adesiva. Foram definidas 36 casas, de 20 cm x 20 cm.

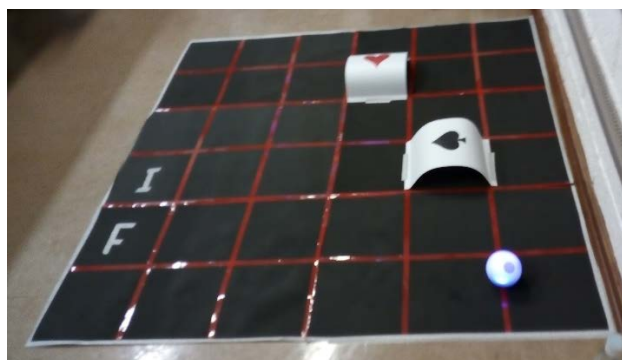


Figura 5: Primeiro protótipo do tabuleiro usado no sistema. Fonte: dos autores.

Este protótipo foi validado com pesquisadores especialistas em IHC (Interação Humano-Computador) em um teste piloto. Neste teste piloto, os pesquisadores, que já eram familiarizados com o TaPrEC em sua versão original, foram convidados a realizar uma atividade (Figura 6) com o sistema completo: TaPrEC, Sphero e tabuleiro. A atividade consistia em locomover o Sphero de uma casa inicial a uma casa final, ambas pré-definidas, de modo que, em seu caminho, a bola robótica passasse sob dois obstáculos em formato de “túnel”. Dois pesquisadores utilizaram o sistema, enquanto que outros dez especialistas observaram o uso.

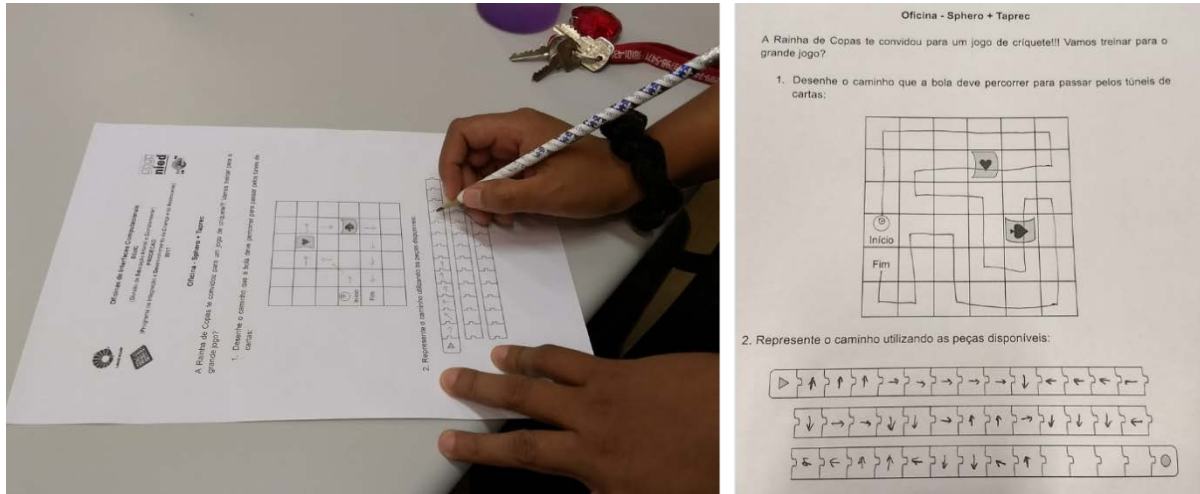


Figura 6: À esquerda, especialista em IHC planejando a atividade; à direita, o exercício proposto em mais detalhes. Fonte: dos autores.

Durante o teste piloto foi possível observar um problema com o tabuleiro utilizado: embora o usuário criasse um programa correto, na execução no mundo físico, o Sphero não fazia o percurso esperado não chegando ao ponto final. Apesar de o problema ser único, pudemos observar três causas diferentes. Foi possível notar que as emendas entre as folhas de EVA e as fitas usadas para delimitar as casas fizeram com que o Sphero, algumas vezes, perdesse velocidade, fazendo com que a bola robótica andasse menos do que o determinado. Também foi observado que o material utilizado era muito fino, permitindo que algo que estivesse na superfície em que foi colocado, interferisse no movimento da bola robótica. Por fim, observamos que a calibração do ângulo inicial e a bateria do Sphero também afetavam os movimentos do dispositivo durante seu caminho.

3.3.2 Segundo Protótipo do Tabuleiro

Na tentativa de resolver o problema encontrado com o primeiro protótipo, construímos uma segunda versão do tabuleiro (Figura 7), com 1,60 m x 0,80 m utilizando duas folhas papelão. O material mais rígido foi escolhido para tentar diminuir o efeito que a superfície no qual o tabuleiro fica apoiado provoca na trajetória do Sphero. Já as emendas e as divisões entre as casas foram feitas com caneta ao invés de fita para evitar que a saliência interferisse no movimento. As casas do tabuleiro foram planejadas com 40 cm x 40 cm, para que, mesmo com um desvio no trajeto, a bola tivesse uma chance maior de chegar ao seu destino.

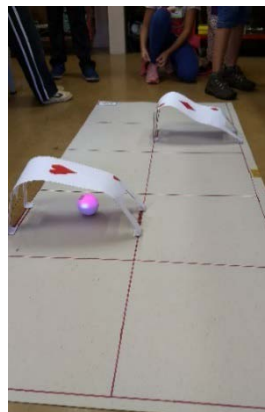


Figura 7: Segundo protótipo do tabuleiro usado no sistema. Fonte: dos autores.



Em síntese, para a integração do TaPrEC com o Sphero foi necessário utilizar uma versão de Raspberry Pi que suportasse conexões Bluetooth. Além disso, foi preciso atualizar o software de controle do sistema para que ele se comunicasse com o Sphero, e criou-se uma camada para abstrair os comandos para movimentar o Sphero em comandos mais simples e fáceis de compreender. Por fim, foi preciso criar um tabuleiro físico, no qual as atividades eram planejadas e executadas. O primeiro protótipo de tabuleiro foi validado com especialistas em IHC, e baseado nos resultados desse experimento, um novo tabuleiro foi criado para uso no ambiente educacional.

4 Estudo Exploratório com TaPrEC + Sphero

O estudo exploratório tem bases qualitativas e utiliza um contexto real de alunos e professores que frequentam e trabalham naquele espaço educacional, utilizando uma sala do tipo “ateliê” para a realização das atividades propostas. As subseções a seguir descrevem os principais aspectos do estudo exploratório em que o ambiente proposto neste trabalho foi avaliado.

4.1 Objetivo

Esse estudo exploratório teve por objetivo avaliar a tangibilidade na etapa de execução dos programas criados com o sistema TaPrEC + Sphero e entender a resposta afetiva dos participantes em relação a sua experiência com o sistema na realização de tarefas envolvendo os conceitos de programação (sequencia, repetição, procedimento).

4.2 Contexto e Participantes

O estudo exploratório foi realizado no 1º semestre de 2017 no espaço ‘ateliê’ do PRODECAD⁶ (Programa de Desenvolvimento e Integração da Criança e do Adolescente) um ambiente que oferece educação complementar em ambiente não-formal para crianças e adolescentes de 6 a 14 anos matriculadas na EEPG “Sergio Pereira Porto”, em horário de contraturno ao ensino regular. O espaço se encontra na Universidade de Campinas (Unicamp), lugar em que essa pesquisa está sendo desenvolvida e os pais das crianças trabalham.

O estudo de caso foi conduzido no formato de 6 Oficinas⁷ práticas, cada uma com duração de aproximadamente 1 hora. Duas Oficinas foram realizadas com 9 professoras dessa instituição com idades entre 33 e 53 anos. Participaram das outras quatro Oficinas 13 crianças que frequentam o PRODECAD, com idades variando de 7 até 10 anos, sendo 7 meninos e 6 meninas.

4.3 Método

A temática das atividades propostas no semestre do estudo era inspirada em “Alice no País das Maravilhas” (Carroll, 1865). Inicialmente as crianças foram introduzidas à obra por meio do filme de mesmo título produzido pela *The Walt Disney Production* em 1951. A temática e os personagens do filme foram utilizados nas atividades preparadas para cada uma das Oficinas.

No início de cada Oficina os participantes eram introduzidos ao objetivo da Oficina e recebiam orientações de como realizar as atividades. Os participantes eram então distribuídos em dois ou três grupos, e deveriam trabalhar em conjunto para completar as tarefas propostas utilizando o ambiente. Na Oficina 3, com as professoras, foi lembrado o ambiente TaPrEC e os conceitos de sequência, procedimento e repetição. As professoras já haviam trabalhado com o ambiente de programação anteriormente (Carbajal e Baranauskas, 2015). Com as crianças, foram

⁶ PRODECAD. <http://www.dgrh.unicamp.br/dedic/prodecad>. Último acesso em maio de 2018.

⁷ Projeto aprovado pelo Comitê de Ética em Pesquisa da Unicamp, sob número 32213314.8.0000.5404.



trabalhados os conceitos de sequência, procedimento e repetição nas Oficinas 1, 2 e 4 respectivamente, usando o ambiente TaPrEC. Na Oficina 5 foram trabalhados com professoras e na Oficina 6 com crianças, os conceitos aprendidos nas Oficinas anteriores, no sistema TaPrEC + Sphero. A Tabela 2 ilustra as Oficinas deste estudo exploratório, com o conceito envolvido e o grupo de participantes. Nos dias 28/03/2017 e 04/04/2017 as professoras realizaram outras atividades relacionadas ao TaPrEC. Em uma das oficinas, as professoras propuseram melhorias para o símbolo utilizado no bloco de programação de repetição. Em outra atividade, elas propuseram exercícios que poderiam utilizar o TaPrEC. Por possuírem focos e objetivos diferentes do deste estudo, essas atividades não foram incluídas neste trabalho.

Tabela 2: Oficinas com o conteúdo trabalhado, grupo participante e se houve ou não o uso do Sphero.

Sistema usado	Data da Oficina	Professoras	Crianças
TaPrEC (execução de maneira virtual)	28/03/2017	-	Oficina 1: Sequência
	04/04/2017	-	Oficina 2: Procedimentos
	11/04/2017	Oficina 3: Sequência, Procedimentos e Repetição	Oficina 4: Repetição
TaPrEC + Sphero (execução no mundo físico)	18/04/2017	Oficina 5: Sequência, Procedimentos e Repetição	Oficina 6: Sequência, Procedimentos e Repetição

Ao final de cada Oficina, os participantes realizavam de maneira individual uma autoavaliação de estados afetivos utilizando como ferramenta o *Self-Assessment Manikin - SAM* (Bradley e Lang, 1994), que consiste em uma representação pictográfica (Figura 8) na qual o participante pode registrar a sua Satisfação, Motivação e Controle em relação ao sistema que está sendo utilizado, selecionando um dos bonecos, ou a posição entre os bonecos, que melhor representa sua satisfação, motivação e controle sobre o sistema. Para cada uma das três dimensões do SAM (representadas nas 3 linhas da matriz), a resposta pode variar de 1 a 9, onde 1 representa o limite inferior e 9 o limite superior da avaliação. Tal instrumento foi escolhido por sinalizar aspectos do engajamento dos participantes no ambiente e na experiência de interação com os elementos do domínio (programação, pensamento computacional).

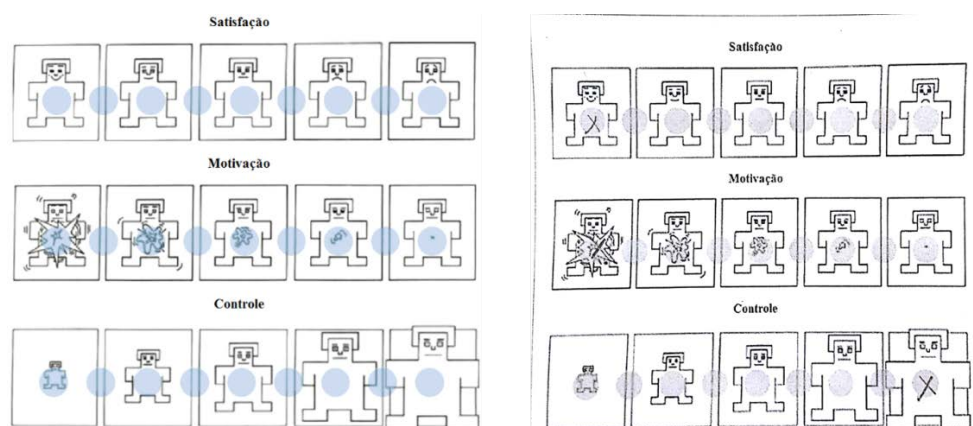


Figura 8: À esquerda, o SAM proposto por Bradley e Lang (1994); à direita, o SAM preenchido por um dos participantes.

As Oficinas também foram registradas por meio de fotos e vídeos, após o consentimento por escrito de todos os participantes ou seus respectivos representantes, no caso de crianças.



A análise desse estudo exploratório contou com as respostas da autoavaliação do estado afetivo de cada participante, a análise das fotos e vídeos e das observações realizadas pelos pesquisadores presentes.

4.4 As Atividades

Inicialmente, nas Oficinas 1 e 3, os grupos tinham que trabalhar o conceito de sequência, utilizando os blocos de “deslocamento” para criar um caminho virtual entre o personagem Alice e algum outro personagem que ela gostaria de convidar para tomar chá da tarde.

Já na Oficina 2, os grupos tinham que utilizar o conceito de sequência e procedimento para fazer o personagem Alice caminhar virtualmente por um trajeto predefinido que estava desenhado na tela do computador. Nessa atividade, cada grupo era responsável por construir um pedaço do caminho, criando para isso um procedimento. Em seguida, as crianças criaram um único programa, combinando cada um dos procedimentos criados pelos grupos e assim fizeram com que o personagem fizesse o trajeto completo.

Na Oficina 4, a atividade pedia que os grupos utilizassem o conceito de repetição para fazer com que um personagem repetisse uma de suas falas do filme por algumas vezes. Para tal, um novo bloco de programação foi criado: quando executado, fazia com que um áudio fosse executado nas caixas de som que foram acopladas ao sistema.


Finalmente, nas Oficinas 5 e 6, cada grupo tinha que completar três atividades (Figura 9) propostas, com diferentes níveis de dificuldade, e baseadas na atividade realizada pelos especialistas em IHC no teste piloto realizado com o primeiro protótipo do tabuleiro (como relatado na seção 3.3.1). As atividades propostas utilizaram a temática do “jogo de críquete da Rainha de Copas” presente no livro “Alice no País das Maravilhas”. Baseado nesse contexto, dois “túneis” em formato de cartas de baralho foram construídos e anexados ao “tabuleiro” do sistema para representar as balizas do jogo de críquete. Dessa maneira, nas atividades propostas, os participantes tinham que mover a bola robótica Sphero pelo “tabuleiro”, de um ponto inicial até um ponto final, obedecendo algumas regras e evitando os obstáculos em formato de cartas. Na primeira atividade, os participantes foram convidados a programar um caminho para que a bola passasse pelas duas balizas. Já na segunda atividade, os participantes tinham que programar o caminho mais curto do ponto inicial até o final, passando novamente pelas duas balizas. Por fim, para retomar o conceito de repetição, na terceira atividade, os participantes deveriam escolher umas das balizas e construir um caminho em que a bola passasse pela baliza duas vezes.

Para cada atividade, em todas as Oficinas, os membros do grupo discutiam e planejavam uma solução em papel (Figura 10a). Em seguida, ainda no papel, os grupos definiram quais blocos de programação seriam utilizados para completar a solução planejada. Por fim, os grupos utilizaram os blocos de programação do ambiente TaPrEC para montar a solução, passaram o leitor RFID sobre cada bloco (Figura 10b). Nas Oficinas 5 e 6, o resultado foi exibido por meio dos movimentos do Sphero no “tabuleiro” (Figura 10c). Nas demais Oficinas, o resultado da execução podia ser acompanhado em um monitor.

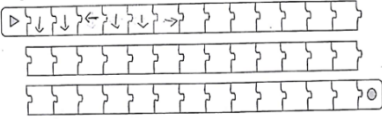


Atividade 1:

1. Desenhe o caminho que a bola deve percorrer saindo do ponto de início, passando pelos túneis de carta e chegando no ponto final:




2. Represente o caminho utilizando as peças disponíveis:

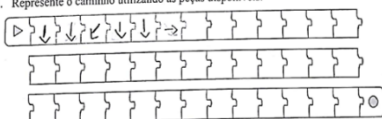


Atividade 2:

1. Desenhe o **menor** caminho que a bola deve percorrer saindo do ponto de início, passando pelos dois túneis de carta e chegando no ponto final:




2. Represente o caminho utilizando as peças disponíveis:



3. Para sua equipe, qual o significado de menor caminho?
 10 caminho ser mais rápido

Atividade 3:

1. Escolha um túnel por onde a bola vai passar. Desenhe o caminho que a bola deve percorrer saindo do ponto de início, passando por um dos túneis duas vezes e chegando ao ponto final.



2. Represente o caminho utilizando as peças disponíveis:

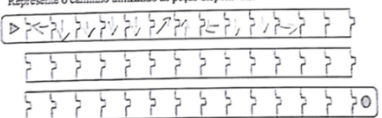


Figura 9: A resolução dos exercícios de um dos grupos de crianças. Fonte: dos autores.

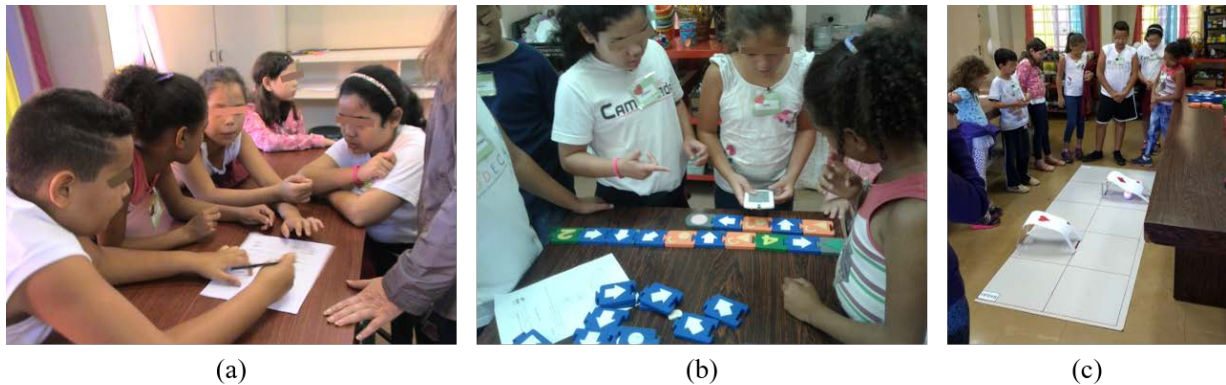


Figura 10: Os passos para o desenvolvimento dos exercícios das Oficinas 5 e 6: (a) planejamento no papel; (b) programação com os blocos tangíveis; (c) execução do programa no mundo físico. Fonte: dos autores.

5 Resultados e Discussão

Conforme objetivos do estudo, queríamos avaliar a ‘experiência’ de interação no ambiente como um todo, incluindo a aprendizagem de conceitos de programação utilizados nas atividades (evidenciados na realização bem-sucedida das atividades propostas) e o engajamento dos participantes no ambiente (evidenciado na percepção dos participantes de sua satisfação, motivação e controle sobre o sistema).



Os formulários de autoavaliação de emoções (SAM) preenchidos pelos participantes em todas as Oficinas foram analisados. Os resultados para as Oficinas com as crianças e professoras podem ser observados na Figura 11 e na Figura 12 respectivamente. As observações dos pesquisadores participantes, bem como os vídeos gravados durante as Oficinas também foram utilizados para complementar a análise do estudo de caso.

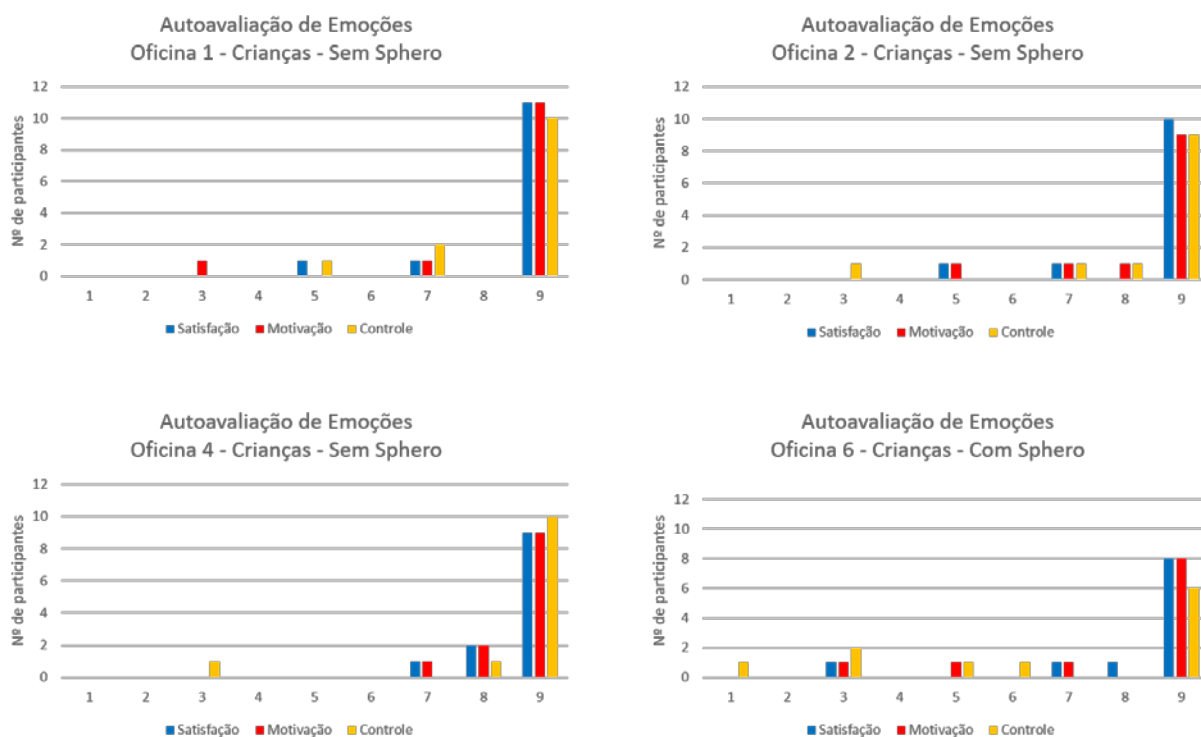


Figura 11: Resultados da autoavaliação do estado afetivo das Crianças. Fonte: dos autores.

O primeiro item avaliado foi a dimensão da “Satisfação” que se refere ao quanto de contentamento e alegria o participante sentiu ao participar das atividades de cada Oficina. Para as crianças, foi possível observar que a moda dos valores apresentados em todas as quatro Oficinas foi 9, o maior valor na escala, indicando que a maioria dos participantes se sentiu totalmente satisfeita ao utilizar o sistema, independentemente do uso do Sphero. Foi possível observar que as crianças ficaram muito empolgadas com a execução no mundo físico da Oficina 6. Por exemplo, toda vez que a execução começava, eles exclamavam frases como “A bolinha está viva!” ou “A bolinha acordou!”. Além disso, as crianças vibravam toda vez que a bola robótica se movia para uma casa do tabuleiro como eles tinham planejado. Elas também comemoraram muito quando a bola chegou ao ponto final.

A “Motivação”, que se refere a quanto o participante estava envolvido com o objetivo da Oficina e quanto estava empolgado em participar dela, foi o segundo item analisado. Novamente, entre as crianças, a moda dos valores foi 9, indicando que a maioria das crianças estava entusiasmada em participar das Oficinas. Um fato que chamou atenção dos pesquisadores na Oficina 6 foi um grupo de estudantes que pediu para ficar durante o intervalo tentando resolver uma das atividades que não tinham conseguido finalizar.

Por fim, foi avaliada a dimensão do “Controle”, que se refere ao domínio do participante em relação ao sistema. Para as crianças, a exemplo das outras duas dimensões, a moda para o Controle também foi 9, indicando que a maior parte das crianças se sentiu no controle do sistema, isto é, perceberam o sistema como de fácil manipulação, independentemente do uso do Sphero. Entretanto, alguns alunos relataram que as atividades propostas na Oficina 6 eram muito difíceis,



o que pode justificar o maior número de avaliações menores que 5 naquela Oficina quando comparada com as demais.

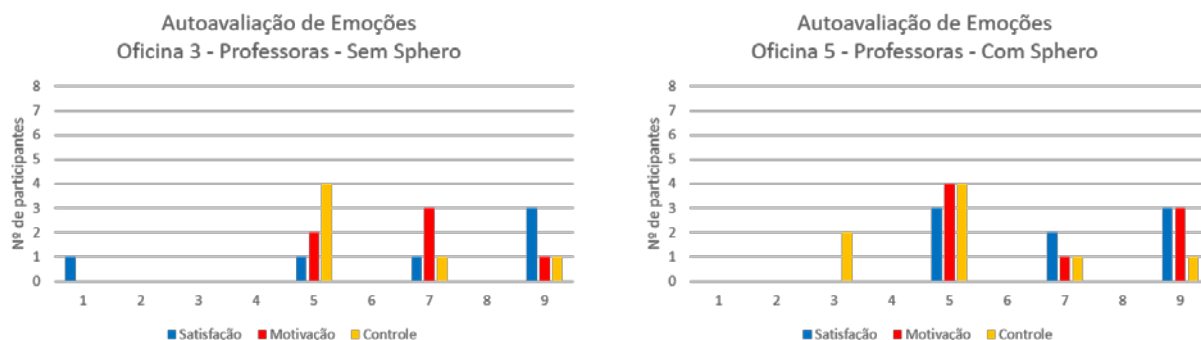


Figura 12: Resultados da autoavaliação do estado afetivo das Professoras. Fonte: dos autores.

Para as professoras, a dimensão “Satisfação” foi positiva em ambas as Oficinas em que elas participaram. Na Oficina 5, em que houve o uso do Sphero, assim como observado com as crianças, pudemos notar que as professoras ficaram empolgadas com a movimentação do Sphero e todas pararam o que estavam fazendo para assistir à execução dos programas criados.

Já em relação ao aspecto “Motivação”, é possível observar na Figura 12 que houve uma redução na moda quando comparado à Oficina 3 (moda 7) e a Oficina 5 (moda 5). Apesar desse resultado no SAM, uma professora relatou que se sentia motivada quando utilizava o sistema TaPrEC + Sphero. Essa professora comentou sobre tentar corrigir a atividade: “Acho que ao mesmo tempo que o sistema é instigante, ele é desafiador: você tem vontade de ver se a bolinha vai andar, se ela vai passar. Isso te instiga, te deixa curiosa para ver se você fez o processo certo ou não. E desafia para fazer de novo para ver se a bolinha vai passar [no lugar certo].”

Por fim, em relação ao aspecto “Controle”, as professoras apresentaram resultados semelhantes entre as duas Oficinas, sendo que as duas tiveram moda 5 para esse aspecto. Entretanto, a Oficina 5, na qual foi utilizado o Sphero, podemos observar algumas notas negativas (entre 1 e 3) para a dimensão “Controle”. Acreditamos que isso se deva a um erro que ocorreu no sistema durante a Oficina 5: em determinada atividade, apesar de o programa escrito estar correto, a execução falhou, e o Sphero não seguiu o caminho definido pelo programa. O mesmo erro aconteceu com o grupo de crianças na Oficina 6. Foi possível observar que ambos os grupos ficaram frustrados e responderam tentando modificar o programa. Nessa situação, os pesquisadores explicaram para os participantes que havia um erro no sistema, pediram para que eles aguardassem enquanto o Sphero era recalibrado e em seguida eles puderam reexecutar seus exercícios, obtendo um comportamento correto da bola robótica.

De maneira geral, foi possível observar que a execução de forma física, que é naturalmente mais lenta do que a virtual, fez com que professoras e alunos conseguissem depurar o programa mais facilmente: como eles acompanham a execução passo a passo, eles conseguiram identificar um erro dentro do programa durante o percurso do Sphero. Apesar da facilidade na depuração, a execução mais lenta pareceu deixar os adultos participantes ansiosos, já que fizeram uma série de perguntas: “Por que que ela [a bolinha] não vai?”, “Por que a bolinha parou?” e “Ela [a bolinha] é devagarzinha?”. Como citado anteriormente, a reação das crianças pareceu ser mais positiva: elas ficaram aguardando cada movimento e vibraram com a movimentação do Sphero casa a casa no tabuleiro.

Observamos que, durante a fase do planejamento do programa, as professoras olhavam para o tabuleiro físico e planejavam mentalmente os movimentos a serem realizados, mesmo sabendo que na folha de exercícios havia uma cópia do tabuleiro, e depois elas passavam os



movimentos planejados para o papel. Quando perguntadas sobre os benefícios da tangibilidade na fase da execução, algumas professoras opinaram. A professora B disse: “É como se fosse um processo de aprendizagem de aquisição de matemática: as crianças pegam alguns objetos para fazer soma. Aqui [no TaPrEC] é a mesma coisa: quando você vê a coisa física é mais fácil, e depois você passa para algo mais abstrato.”. Já a professora C complementou: “Quando eu me levanto e olho o concreto [o tabuleiro] parece que abre todo um processo de compreensão para fazer no papel.”. Ainda, uma professora sugeriu que o sistema com execução no mundo físico seria benéfico para o aprendizado das crianças: “Achei bem mais divertido e acho que a criança gosta muito quando ela pode brincar, vendo a bolinha andar, e não só no computador.”.

O estudo apresentado tem bases qualitativas e é situado. Como tal, sua validade é baseada nas evidências de resultados analisados da resposta afetiva dos participantes (alunos e professores), observações dos pesquisadores *in loco* e *a posteriori* pela análise de depoimentos registrados em fotos e vídeos. Tais resultados apontam para um engajamento efetivo principalmente das crianças no ambiente e a realização bem-sucedida de tarefas envolvendo os conceitos de programação, que compõem a sua experiência como um todo.

6 Conclusão

A literatura tem mostrado, há algum tempo, a relevância de introduzir as crianças à programação em seu ambiente escolar, para que elas possam se beneficiar do pensamento computacional como ferramenta em outros domínios. Vários trabalhos têm mostrado também que quando um sistema tangível é utilizado para o ensino de programação, a aprendizagem é facilitada. A tangibilidade tem sido mais associada à construção propriamente dita dos programas. Este trabalho explorou a tangibilidade também na execução de programas, ao expandir o TaPrEC, um ambiente de programação tangível, para incluir o Sphero, como elemento tangível, uma tecnologia contemporânea que responde aos comandos do TaPrEC de maneira concreta e lúdica.

Os resultados do estudo exploratório conduzido com o ambiente tecnológico proposto em contexto educacional real, ilustrados na avaliação da satisfação, motivação e controle na realização das atividades, mostraram a efetividade do ambiente no engajamento dos participantes com a atividade e no fazer sentido dos conceitos abordados.

A expansão de possibilidades que as tecnologias contemporâneas (baseadas em sensores e atuadores) podem trazer ao conceito de tangibilidade e ao nível de experiência dos usuários com os conceitos da programação e do pensamento computacional abrem novas avenidas de pesquisa. Novos desdobramentos de ambientes tangíveis que aportem novas tecnologias e explorem habilidades do pensamento computacional sob novas perspectivas são possíveis. Por exemplo, um trabalho em andamento propõe a investigação sobre o uso da Internet das Coisas (IoT) em conjunto com ambientes tangíveis em contexto educacional. Outros trabalhos têm explorado o conceito de aprendizagem criativa apoiada por tecnologia vestível, programação tangível e robótica.

Agradecimentos

Este trabalho teve apoio do GGBS, Grupo Gestor de Benefícios Sociais da UNICAMP, do CNPq (#306272/2017-2 e #140536/2019-1), Fapesp (#2015/16528-0 e #2015/24300-9) e CAPES (#1545149/2015 e #1654045/2016). Agradecemos também aos professores e alunos do PRODECAD, ao NIED (Núcleo de Informática Aplicada a Educação) da UNICAMP e aos pesquisadores do grupo InterHAD (*Human-Digital Artifact Interaction*).



Referências

- Baranauskas, M. C. C. e Carbajal, M. L. (2017). The Social Nature of Programming: Children and Fluency. In: Kurosu M. (eds) Human-Computer Interaction. Interaction Contexts. HCI 2017. Lecture Notes in Computer Science, vol 10272. Springer, Cham. doi: [10.1007/978-3-319-58077-7_23](https://doi.org/10.1007/978-3-319-58077-7_23). [GS Search]
- Baranauskas, M. C. C. e Posada, J. E. G. (2017). Tangible and Shared Storytelling: Searching for the Social Dimension of Constructionism. In Proceedings of the 2017 Conference on Interaction Design and Children (IDC '17). ACM, New York, NY, USA, 193-203. doi: [10.1145/3078072.3079743](https://doi.org/10.1145/3078072.3079743). [GS Search]
- Bradley, M. M. e Lang, P. J. (1994). Measuring Emotion: the self-assessment manikin and the semantic differential. In Journal of behavior therapy and experimental psychiatry, 25 (1), 49-59. [GS Search]
- Carbajal, M. L. e Baranauskas, M. C. C. (2015). TaPrEC: Desenvolvendo um Ambiente de Programação Tangível de Baixo Custo para Crianças. In Nuevas Ideas en Informática Educativa TISE. 363-370. [GS Search]
- Carroll, L. (1865). Alice's Adventures in Wonderland. MacMillan. [GS Search]
- Fan, M., Antle, A. N. e Cramer, E. S. (2016). Design Rationale: Opportunities and Recommendations for Tangible Reading Systems for Children. In Proceedings of the 15th International Conference on Interaction Design and Children (IDC '16). ACM, New York, NY, USA, 101-112. DOI: [10.1145/2930674.2930690](https://doi.org/10.1145/2930674.2930690). [GS Search]
- Ishii, H. e Ullmer, B. (1997) Tangible bits: towards seamless interfaces between people, bits and atoms. In Proceedings of the ACM SIGCHI Conference on Human factors in computing systems (CHI '97). ACM, New York, NY, USA, 234-241. doi: [10.1145/258549.258715](https://doi.org/10.1145/258549.258715). [GS Search]
- Ishii, H. (2008). Tangible bits: beyond pixels. Proceedings of the 2nd international conference on Tangible and Embedded Interaction (TEI '08). ACM, New York, NY, USA, xv-xxv. doi: [10.1145/1347390.1347392](https://doi.org/10.1145/1347390.1347392). [GS Search]
- Liao, Y. K. C. e Bright, G. W. (1991). Effects of Computer Programming on Cognitive Outcomes: A meta-analysis. Journal of Educational Computing Research, 7(3), 251-268. doi: [10.2190/E53G-HH8K-AJRR-K69M](https://doi.org/10.2190/E53G-HH8K-AJRR-K69M). [GS Search]
- Moreira, E. A., Dos Reis, J. C., Baranauskas, M. C. C. (2019). Tangible Artifacts and the Evaluation of Affective States by children. Brazilian Journal of Computers in Education, v. 27, n. 01, p. 58. doi: [10.5753/rbie.2019.27.01.58](https://doi.org/10.5753/rbie.2019.27.01.58). [GS Search]
- Panaggio, B. Z. e Baranauskas, M. C. C. (2017). Explorando as Possibilidades do Sphero em um Ambiente Educacional. Em Anais do XXIII Workshop de Informática na Escola (WIE 2017). 245-254. doi: [10.5753/cbie.wie.2017.245](https://doi.org/10.5753/cbie.wie.2017.245). [GS Search]
- Papert, S. (1980). Mindstorms: Children, Computers, and Powerful Ideas. New York: Basic Books. [GS Search]
- Posada, J. E. G., Hayashi, E. C. S. e Baranauskas, M. C. C. (2014). On Feelings of Comfort, Motivation and Joy that GUI and TUI Evoke. In Design, User Experience, and Usability. User Experience Design Practice. 273-284. Springer International Publishing. doi: [10.1007/978-3-319-07638-6_27](https://doi.org/10.1007/978-3-319-07638-6_27). [GS Search]



- Prensky, M. (2001). Digital Natives, Digital Immigrants. *On the Horizon*, Vol. 9 No. 5, 1–6. doi: [10.1108/10748120110424816](https://doi.org/10.1108/10748120110424816). [GS Search]
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. e Kafai, Y. (2009). Scratch: Programming for All. *Commun. ACM* 52, 11 (November 2009), 60-67. doi: [10.1145/1592761.1592779](https://doi.org/10.1145/1592761.1592779). [GS Search]
- Rusk, N., Resnick, M., Berg, R. e Pezalla-Granlund, M. (2008). New Pathways into Robotics: Strategies for broadening participation. *Journal of Science Education and Technology*, 17(1), 59-69. [GS Search]
- Sabourin, J., Kosturko, L. e McQuiggan, S. (2016). CodeSnaps: Block-Based Robotic Programming for the Low-Budget Classroom. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16)*. ACM, New York, NY, USA, 242-242. doi: [10.1145/2839509.2850508](https://doi.org/10.1145/2839509.2850508). [GS Search]
- Sánchez, I., Cortés, M., Riekkki, J. e Oja. M. (2011). NFC-based interactive learning environments for children. In *Proceedings of the 10th International Conference on Interaction Design and Children (IDC '11)*. ACM, New York, NY, USA, 205-208. doi: [10.1145/1999030.1999062](https://doi.org/10.1145/1999030.1999062). [GS Search]
- Shaer, O. e Hornecker, E. (2010). Tangible User Interfaces: past, present, and future directions. *Foundations and Trends in Human-Computer Interaction*, vol. 3, no. 12, 1-137. doi: [10.1561/11000000026](https://doi.org/10.1561/11000000026). [GS Search]
- Sipitakiat, A. e Nusen, N. (2012). 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)*. ACM, New York, NY, USA, 98-105. doi: [10.1145/2307096.2307108](https://doi.org/10.1145/2307096.2307108). [GS Search]
- Suzuki, H. e Kato, H. (1995). Interaction-level Support for Collaborative Learning: AlgoBlock—an open programming language. In *The first international conference on Computer support for collaborative learning (CSCL '95)*, John L. Schnase and Edward L. Cunniss (Eds.). L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 349-355. doi: [10.3115/222020.222828](https://doi.org/10.3115/222020.222828). [GS Search]
- Sylla, C., Branco, P., Coutinho, C. e Coquet, E. (2012). TUIs vs. GUIs: comparing the learning potential with preschoolers. *Personal and Ubiquitous Computing*, Springer, Vol 16, No. 4, 421-432. doi: [10.1007/s00779-011-0407-z](https://doi.org/10.1007/s00779-011-0407-z). [GS Search]
- Trower, J. e Gray, J. (2015). Blockly Language Creation and Applications: Visual Programming for Media Computation and Bluetooth Robotics Control. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15)*. ACM, New York, NY, USA, 5-5. doi: [10.1145/2676723.2691871](https://doi.org/10.1145/2676723.2691871). [GS Search]
- Valente, J. A., Freire, F. M. P., Arantes, F. L., Viegas d'Abreu, J. V., Amiel, T. e Baranauskas, M. C. C. (2017). Alan Turing tinha Pensamento Computacional? Reflexões sobre um campo em construção. *Tecnologias, Sociedade e Conhecimento*, 4(1), 7-22. [GS Search]
- Wang, D., Qi, Y. e Zhang, L. (2015). A Tangible Programming System Conveying Event Handling Concept. In *Proceedings of the 14th International Conference on Interaction Design and Children (IDC '15)*. ACM, New York, NY, USA, 319-322. doi: [10.1145/2771839.2771906](https://doi.org/10.1145/2771839.2771906). [GS Search]



- Wang, D., Wang, T. e Liu, Z. (2014). A Tangible Programming Tool for Children to Cultivate Computational Thinking. *The Scientific World Journal*, vol. 2014, Article ID 428080, 10 pages. [[GS Search](#)]
- Wing, J. M. (2006). Computational Thinking. *Commun. ACM* 49, 3 (March 2006), 33-35. doi: [10.1145/1118178.1118215](https://doi.org/10.1145/1118178.1118215). [[GS Search](#)]
- Wing, J. M. (2014). Computational thinking benefits society. 40th Anniversary Blog of Social Issues in Computing. [[GS Search](#)]
- Zuckerman, O., Arida, S. e Resnick, M. (2005). Extending Tangible Interfaces for Education: digital montessori-inspired manipulatives. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 859-868). ACM. doi: [10.1145/1054972.1055093](https://doi.org/10.1145/1054972.1055093). [[GS Search](#)]