

## **Desenvolvimento de exergames por professores não programadores: uma metodologia ativa de aprendizagem para o ambiente *scratch***

*Luciano Kercher Greis\**

*Kamyla Thais Dias de Freitas\*\**

*Fernando Luiz Cardoso\*\*\**

### **Resumo:**

Apontado como recurso positivo, o uso de exergames no ambiente educacional encontra-se normalmente restrito a utilização de aplicações comerciais ou ao complexo desenvolvimento de produtos que frequentemente não possui acompanhamento de profissionais especializados. Contudo, a utilização de recursos tecnológicos acessíveis, como o ambiente de programação Scratch, possibilita que qualquer pessoa desenvolva um exergame simplificado, sem a necessidade de conhecimentos avançados sobre programação e informática. O objetivo deste artigo é descrever os pressupostos teóricos de uma metodologia, apoiado nos conceitos construtivistas de Piaget, para propor que professores não programadores construam exergames adaptados às suas necessidades. A aplicação desta metodologia dar-se-á posteriormente em uma oficina, a qual se pretende envolver o professor na construção dos conceitos computacionais necessários para a atividade. Ao final, esperamos proporcionar autonomia suficiente ao professor para que possa expressar-se tecnologicamente, criando seus próprios exergames.

**Palavras chave:** Exergames, Formação de professores, Metodologia ativa.

\* Universidade do Estado de Santa Catarina. Florianópolis, Santa Catarina, Brasil.  
E-mail: lucianokgsl@gmail.com. <http://orcid.org/0000-0001-9050-0014>.

\*\* Universidade do Estado de Santa Catarina. Florianópolis, Santa Catarina, Brasil.  
E-mail: Kamyla.freitas@outlook.com. <http://orcid.org/0000-0002-0770-7952>.

\*\*\* Universidade do Estado de Santa Catarina. Florianópolis, Santa Catarina, Brasil.  
E-mail: fernandocardoso.ph.d.lagesc@gmail.com. <http://orcid.org/0000-0002-3074-0988>.

## Desarrollo de exergames por profesores no programadores: una metodología activa de aprendizaje para el ambiente *scratch*

### Resumen

Apuntado como recurso positivo, el uso de exergames en el ambiente educativo se encuentra normalmente restringido al uso de aplicaciones comerciales o al complejo desarrollo de productos que no tiene acompañamiento de profesionales especializados. Sin embargo, la utilización de recursos tecnológicos accesibles, como el entorno de programación Scratch, permite a cualquier persona desarrollar un exergame simplificado, sin la necesidad de conocimientos avanzados sobre programación e informática. El objetivo de este artículo es describir los presupuestos teóricos de una metodología, apoyado en los conceptos constructivistas de Piaget, para proponer que profesores no programadores construyan exergames adaptados a sus necesidades. La aplicación de esta metodología se dará posteriormente en un taller, la cual se pretende involucrar al profesor en la construcción de los conceptos computacionales necesarios para la actividad. Al final, esperamos proporcionar autonomía al profesor para que pueda expresarse tecnológicamente, creando sus exergames.

**Palabras clave:** Exergames, Formación de profesores, Metodología activa.

## Development of exergames by non-programming teachers: an active methodology of learning for the scratch environment

### Abstract

Pointed as a positive resource, the use of exergames in the educational environment is usually restricted to the use of commercial applications, or to the complex and not always possible development of products by specialized professionals. However, using accessible technological resources such as the *Scratch* programming environment today enables anyone to develop a simplified exergame without the need for advanced programming and computer skills. The purpose of this article is to describe the theoretical assumptions of a methodology based on the constructivist concepts of Piaget, to propose that non-programmers construct their own exergames autonomously, adapted to their needs. The application of this methodology will be given later in a workshop, which intends to involve the teacher actively in the construction of the computational concepts necessary for the activity. At the end of this workshop, we hope to provide the teacher with sufficient autonomy so that he can express himself technologically, creating his own exergames.

**Key words:** Exergame, Teacher training, Active methodology.



## Introdução

De forma simplificada, exergames são qualquer tipo de videogames/interações multimídia, que exigem do jogador movimentar-se fisicamente para jogar (OH; YANG, 2010). Esta possibilidade de interação com a interface do jogo a partir de tecnologias que permitem o rastreamento de movimentos do corpo do participante, retirou do videogame o estigma de ser uma atividade exclusivamente sedentária, oferecendo uma ampla gama de aplicações nas áreas da saúde e educação. Neste sentido, autores apontam para a possibilidade de utilização dos exergames como alternativa ou complemento a prática de exercícios no âmbito da Educação Física curricular (FINCO et al., 2015), como ferramentas de reabilitação, aprendizagem e intervenção motora (ANDERSON; ANNETT; BISCHOF, 2010), entre outros.

Entretanto, as práticas descritas na literatura invariavelmente estão restritas à utilização de produtos comerciais, ou seja, jogos que foram concebidos para fins de entretenimento. Isto remete a necessidades de adaptações de uso, que podem não ocorrer de forma simples. Jogos comerciais possuem reduzidas possibilidades de customização de suas variáveis, podendo levar muitas vezes a impossibilidade do professor seguir sua própria metodologia, ou ainda, impossibilitar o alcance de todos os objetivos inicialmente propostos para a atividade pedagógica a partir de seu uso.

Mesmo trazendo requisitos necessários para propor engajamento dos alunos (CHAPMAN, 2003), estes jogos comerciais exigem do profissional a necessidade de contornar eventuais narrativas indesejadas, complexos sistemas de recompensas, desafios, estados de vitória, combates, regulação do nível de exigência, entre outros elementos da mecânica de jogo que foi proposta exclusivamente para diversão, e que fogem do controle, ou mesmo do interesse do profissional que pretende fazer uso daquele recurso.

Como alternativa, o desenvolvimento de produtos específicos remete constantemente ao problema da criação de jogos digitais, que por mais simples que possam ser os requisitos de uso, está normalmente associada à percepção de dificuldade em utilizar alguma linguagem de programação muito complexa. Também, a produção comercial envolve normalmente não um, mas diversos profissionais especializados para o seu desenvolvimento, tornando esta, uma opção proibitiva na maioria dos casos, pelo alto custo e complexidade de implementação.

Uma terceira via surge a partir da utilização de ambientes de programação simplificados, baseados em interfaces gráficas. O ambiente de programação *Scratch*, que será utilizado nas oficinas, constitui-se de uma linguagem de programação visual acessível a todas as pessoas, e permite a criação de objetos interativos ricos em recursos de mídia (MARJI, 2014, p. 17). Ao oferecer conectividade facilitada com o sensor *Kinect* (controle desenvolvido para o console de videogame *Microsoft Xbox 360* ou *One*), este ambiente passa a permitir também o desenvolvimento de exergames de forma simplificada.

Alunos de todas as idades já fazem uso em atividades escolares, no ambiente *Scratch* ou similares, produzindo histórias animadas, jogos e simulações, no contexto de disciplinas como Matemática (ANDRADE; SILVA; OLIVEIRA, 2013), Computação (SCAI-CO, 2012), produção textual (SILVA; MONTANÉ, 2016), entre outros. Igualmente, são



encontradas evidências da sua utilização no ensino superior como recurso inicial em diversos cursos ligados à informática e programação (MÉLO et al., 2011).

Ao propor o desenvolvimento de exergames pelos próprios professores, adaptados à sua realidade e sua metodologia de ensino, buscamos aproximar o professor, ainda resistente aos benefícios anteriormente citados, para o uso de exergames na prática educacional, pois esta será em última análise, a sua própria prática pedagógica, utilizando um recurso que lhe permite expressar-se tecnologicamente, e que traz novas possibilidades educacionais. Também consideramos que os produtos, desenvolvidos pelos próprios professores, estarão em conformidade com a metodologia que professa com as expectativas do currículo que segue, e com as especificidades dos estudantes e ambientes nos quais serão utilizados.

A escolha pelas ferramentas descritas nesta abordagem seguiu alguns pré-requisitos, que incluem a liberdade de produção que o ambiente oferece e os níveis reduzidos de conhecimentos necessários prévios em informática. A necessidade de aprender as especificidades necessárias para o uso do ambiente *Scratch*, as possibilidades e limitações da interação deste ambiente com o sensor *Kinect* e as possibilidades pedagógicas de utilização de exergames por professores que não possuem conhecimentos básicos em programação, nos conduziu a propor uma oficina de desenvolvimento de exergames, que iremos descrever neste artigo. A metodologia utilizada segue alguns dos pressupostos já descritos na literatura por autores como Piaget (1977), Resnick (2007), e Marji (2014), e parte da proposta de um recurso pedagógico a ser descrito neste artigo, um jogo de tabuleiro. O objetivo passa a ser descrever uma metodologia que possibilite aos professores não programadores, expressarem-se tecnologicamente no desenvolvimento de exergames com o uso do ambiente *Scratch*.

### *Scratch*

O ambiente de programação *Scratch*, foi proposto segundo Resnick (2007), a partir do conceito de pensamento computacional (*Computational Thinking*), ou um “conjunto de estratégias, habilidades e capacidades de desenhar sobre ideias do mundo da computação, tais como abstração, depuração, problematização e decomposição”. As características desta linguagem de programação foram pensadas e desenvolvidas pelo *Lifelong Kindergarten Group*, grupo de pesquisa do *MIT Media Lab*, com o propósito de introduzir a programação de maneira fácil e rápida às pessoas de todas as idades e que não possuem nenhum tipo de experiência no assunto (MALONEY et al. 2010).

Com base na metáfora de um jogo de blocos de encaixar, a construção do programa no ambiente *Scratch* se dá pela associação de blocos lógicos, os quais possuem um comando descrito em sua superfície e ao serem conectados constroem uma expressão. O desafio recai sobre a necessidade de aprender alguns conceitos específicos que tornam a simples indicação e uso deste ambiente e seus recursos, insuficiente para que a maioria dos professores o utilizem como uma ferramenta para a construção de jogos ou aplicativos.

A associação com outros recursos tecnológicos tem se tornado bastante comum na comunidade de utilizadores do *Scratch*, e as possibilidades passam pela associação com mundos virtuais (*Second Life* e *Opensim*); pela associação com hardwares (*Arduino*),



largamente utilizado nos processos de automação por hardware e robótica educacional; associação com plataformas de redes sociais (*Twitter*), entre outras. Da mesma forma, a utilização de alguns softwares adicionais de distribuição gratuita, permitem a associação com o sensor de movimento do videogame *Microsoft Xbox* nas versões *360* ou *One* (HOWELL, 2012), possibilitando desenvolver a partir de sua plataforma exergames personalizados.

### *Aprendizagem*

Toda a obra de Piaget pode ser entendida como a procura de resposta para uma pergunta: “Como um sujeito passa de um conhecimento mais simples a um conhecimento mais complexo?” (BECKER, 2012, p. 71). Nesta abordagem, a aprendizagem pode ser definida como a construção de novidades por meio da relação entre o já sabido e o novo.

Becker (2011, p. 35) afirma que Piaget buscava nos conceitos de assimilação e acomodação de estruturas operatórias, “a explicação desta construção inventiva”, destas novidades. Neste sentido, a assimilação consiste na incorporação de elementos novos a uma estrutura já existente, e a acomodação refere-se às transformações sofridas pela estrutura para incorporar os novos elementos. O conceito de estruturas remete a organização ou reorganização dos raciocínios/conhecimentos que o sujeito possui, os quais podem ser adquiridos por descoberta, ou criados pelo próprio sujeito.

Para o autor, o ponto de partida é uma novidade que faça sentido e seja capaz de causar um desequilíbrio nas estruturas operatórias já existentes no sujeito. Ao buscar novamente o equilíbrio, a partir de movimentos de assimilação e a acomodação em seus sistemas de significação, seus sistemas lógicos, o aprendiz pode realizar a abstração reflexiva, ou a tomada de consciência necessária à generalização do conceito. Com o aumento das generalizações, logo, o aumento da extensão do conceito, aumenta a compreensão, possibilitando a transposição do conhecimento a outras situações. A este movimento de equilibração entre a assimilação e a acomodação, Piaget denominou adaptação.

Marques (2008, p. 21) afirma que “as estruturas já construídas anteriormente garantem a continuidade, mas elas são reformuladas, em função de novas assimilações”. Na concepção construtivista, o importante “é a capacidade do sujeito de estabelecer relações entre o novo e o velho, a capacidade para assimilar o novo e acomodar, ou seja, transformar o velho, produzindo sínteses indefinidamente renovadas”. Portanto, “uma aprendizagem não parte jamais do zero” (PIAGET; GRÉCO, 1974, p. 69). Pressupõe-se que novas informações, desafios ou novidades, sejam relacionados aos conhecimentos que o aluno já possui, gerando condições para que se construam novos conhecimentos. E nesta perspectiva a função do professor é “inventar situações experimentais para facilitar a invenção de seu aluno” (Becker, 2003, p.13).

Piaget e Gréco (1974, p. 52-54), também distinguem dois tipos de aprendizagem: no sentido restrito (*stricto sensu*), “só falaríamos de aprendizagem na medida em que um resultado (conhecimento ou atuação) é adquirido em função da experiência”. A aprendizagem no sentido amplo (*lato sensu*) refere-se à união das aprendizagens *stricto sensu* com os processos de equilibração. “Trata-se nesse caso de compreensões graduais e não mais imediatas, mas tais que o confronto de duas afirmações a e b [...] dá lugar a um



resultado novo c”. Este resultado alcançado, c, é proveniente do fato de que uma contradição entre a e b foi levantada, enquanto, até então não tinha sido percebida ou não pode ser ultrapassada.

Trazendo para o contexto da oficina descrita neste trabalho, o participante atinge uma aprendizagem *stricto sensu* quando ele reconhece e consegue construir as atividades iniciais que são ensinadas. A aprendizagem *lato senso* ocorre quando se consegue construir qualquer atividade, por mais desconhecida que seja (mesmo que possa, é claro, conter alguns erros). Esta perspectiva, que se refere ao próprio processo de letramento, é que procuramos para uma oficina de construção de exergames.

### ***Jogo de Tabuleiro***

O ponto inicial desta proposta é o jogo de tabuleiro, construído pelo autor do projeto especificamente para esta oficina, utilizamos um material emborrachado (emborrachado vinil acetílico – E.V.A.), recortado em formato similar aos blocos de programação que encontramos no ambiente *Scratch*. Também foram confeccionados três tabuleiros impressos em papel contendo desafios diferentes, um conjunto de cartas que permite a inserção de variáveis aos blocos de programação, e um personagem.

A atividade engloba quatro desafios de movimentação do personagem pelo tabuleiro do jogo, a partir da construção de um comando (programação) formado pela combinação destes blocos em E.V.A. pelos jogadores. Ao mesmo tempo que buscamos nesta simplicidade de apresentação, que ela faça sentido para todos os participantes independente do conhecimento prévio dos mesmos, buscamos também reconstruir parte da vivência de uso do ambiente *Scratch* em um formato físico.

Por meio destes desafios, novos elementos e situações surgem, procurando causar os desequilíbrios necessários a reconstrução de algumas das estruturas prévias dos sistemas de significado dos participantes. Esta experiência envolve além dos conceitos de programação por blocos e da interface (específicos do *Scratch*), a construção de cenários, personagens, narrativas de jogos, elementos de colaboração e competição, entre outras possibilidades (Figura 1):

Figura 1. Jogo de tabuleiro desenvolvido pelo autor



O primeiro desafio proposto para a atividade consiste em guiar o personagem em linha reta por 8 casas até alcançar o ponto “X”, indicado no tabuleiro. Para isto, o jogador deve posicionar o bloco lógico “Quando executar”, um evento, que indica que a ação encaixada neste bloco será realizada pelo personagem. Logo abaixo, o jogador encaixa o bloco lógico “Andar”, a ação, e insere a carta de variáveis contendo o número “8”, indicando que o personagem deve andar oito casas no tabuleiro.

Becker (2012, p. 37) afirma que para conseguir êxito na construção de estruturas:

o sujeito tem que se apropriar de suas ações; primeiramente de seus esquemas, depois das coordenações de seus esquemas ou coordenações de suas ações; mais adiante dos subsistemas de esquemas, assimilando-os uns aos outros. É esse o caminho da formação de estruturas lógico-matemáticas com as quais poderá apropriar-se dos conhecimentos da ciência lógica e da ciência matemática e, mais tarde, fazer lógica, matemática ou qualquer outra ciência.

No contexto desta atividade, ao agir sobre os elementos físicos do jogo para solucionar os problemas dispostos no tabuleiro, é necessário que o professor realize determinadas abstrações acerca da coordenação de suas ações, ou seja, aproprie-se de suas ações. É a coordenação de estruturas de ação para Piaget (BECKER, 2011, p. 51), a chave para a aprendizagem. À medida em que os processos de assimilação das novidades apresentadas pelos desafios propostos no jogo geram desequilíbrio, isto é, trazem problemas que precisam ser resolvidos pelos participantes, as aquisições são conservadas como complexas organizações, exigindo que o jogador se adapte a estas novidades, alternando ou construindo novas estruturas de assimilação.

Ainda, o autor afirma ser “impossível tratar da aprendizagem sem tratar dos esquemas” (BECKER, 2011, p. 59). Os esquemas constituem a condição de possibilidade (condição necessária) de toda a aprendizagem *stricto sensu*, e o processo de construção de novos esquemas e de sua coordenação é o próprio processo de equilíbrio, ou seja, o processo de aprendizagem *lato senso*.

Os blocos de programação construídos para esta atividade (Figura 2), constituem propositalmente uma aproximação do formato dos blocos de programação que envolvem a atividade também no ambiente digital. É a partir da manipulação física destes blocos, na tentativa de encaixá-los de uma forma que faça sentido, que leva o sujeito a abstrair, inicialmente de forma empírica (BECKER, 2012, p. 35-36), apoiando-se “sobre observáveis dos objetos e das ações nas suas características materiais”, ou seja, sobre o formato de encaixe das peças, as expressões descritas e as variáveis que ele pode selecionar nos blocos de programação que compõe a atividade.

Posteriormente, os jogadores irão retirar desta interação com os blocos de programação, não suas características empíricas (encaixe, expressões, variáveis), mas o que Piaget chama de pseudoempíricas (BECKER, 2012 p. 36), ou as características “que o sujeito colocou neles – quando o sujeito descobre, nos objetos, propriedades introduzidas neles por sua atividade”, tais como a capacidade de criar um comando para fazer o personagem avançar quatro casas à frente, e girar para a esquerda, e repetir este comando três vezes para atingir o objetivo. Destas interações, resultam as possibilidades de associação entre os blocos, para formar comandos iniciais.



Figura 2. Blocos de programação



A abstração reflexionante refletida (BECKER, 2012, p. 36), é uma abstração que se transformou por tomada de consciência. “É essa tomada de consciência que faz surgir os conceitos, sem os quais não podemos pensar; eles são a condição de possibilidade do nosso pensar”. É a partir da elaboração destes conceitos no sujeito, e que não pode ser “transmitido”, mas deve ser construído pelo processo de aprendizagem, que o leva a desenvolver ou inventar os novos programas, os novos desafios, mesmo que não tenham sido ensinados previamente para ele.

Toda a ação do sujeito sobre a organização dos blocos de programação busca um objetivo, que pode resultar em êxito ou fracasso, mas que são constatações conscientes. “A busca do motivo do fracasso leva o sujeito a tomar consciência das regiões mais centrais da ação” (BECKER, 2011, p. 165). Após verificar no objeto um resultado frustrado, o sujeito vai buscar saber no que o esquema não se adaptou ao objeto, procurando fazer as devidas correções.

Para a dinâmica do tabuleiro, foram desenvolvidos dois conjuntos idênticos de comandos semelhantes ao formato encontrado no ambiente *Scratch*, e elaborados quatro desafios de complexidade crescente. A dinâmica inicia dividindo-se o número de participantes (proposto para dez participantes) entre dois grupos que competem entre si, ao mesmo tempo em que seus integrantes colaboram uns com os outros dentro do grupo, para tentar solucionar o mais rápido possível as atividades.

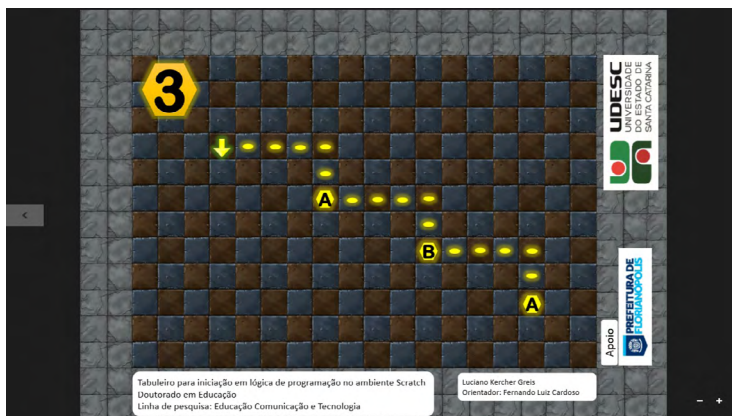
Estas atividades consistem em fazer o personagem sair do ponto inicial indicado por uma flecha, seguir a trilha demarcada em amarelo nos tabuleiros, e a partir do segundo desafio, acionar o brinquedo “osso” caso cruze por uma letra “A”, ou ainda acionar o brinquedo “galinha” caso a letra seja um “B”. Ambos os brinquedos possuem uma letra associada e um apito, e ganha a prova o grupo que acionar primeiro este apito de forma correta. O código construído também deve estar descrito corretamente, caso contrário a equipe perde o ponto.

Uma alternativa a utilização dos tabuleiros (Figura 3) surge ao se propor utilizar a própria marcação do piso (lajotas) como cenário, sendo que um dos participantes do grupo incorpora o “Robô”, personagem fictício da dinâmica, e somente poderá seguir os passos ditados por um outro participante que lê o “software” construído pelo restante do grupo, unindo os comandos e variáveis em E.V.A. para solucionar o desafio. Assim como



no ambiente *Scratch*, as peças do jogo possuem um encaixe específico, e somente podem ser unidas se a construção lógica estiver correta.

Figura 3. Tabuleiro impresso para a atividade



Além de introduzir os conceitos computacionais iniciais e a lógica de construção dos códigos de programação contida no ambiente *Scratch*, esta dinâmica proporciona o debate acerca de vários pontos que permeiam a construção dos jogos na oficina proposta, tais como conceitos de competição e colaboração, construção de narrativas digitais, jogos cognitivos, didática escolar, games, entre outros.

### ***Proposta de Oficina***

O objetivo proposto para esta oficina consiste em que o participante apresente ao final do período um produto finalizado, construído de forma autônoma, de acordo com suas necessidades e metodologias próprias. Organizado inicialmente para dez encontros com duração de quatro horas cada, reservamos os quatro encontros iniciais para tratar dos recursos necessários para a aprendizagem do ambiente *Scratch*, sendo um período comum a todos os participantes. Os seis encontros seguintes são para que os professores iniciem seus projetos individuais, desenvolvam, testem, tirem dúvidas, compartilhem e contribuam nos processos dos demais colegas envolvidos nas atividades.

Desta forma, os recursos utilizados nos encontros estão divididos da seguinte maneira:

Quadro 1. Organização dos conteúdos da oficina.

1º Encontro	<ul style="list-style-type: none"> <li>- Exposição inicial do que trata a formação</li> <li>- Dinâmica utilizando o jogo de tabuleiro</li> <li>- Explorar o ambiente Estudo <i>Code</i></li> <li>- Explorar a biblioteca <i>Scratch</i></li> </ul>
2º e 3º Encontro	- Primeiros passos utilizando <i>Scratch</i>
4º Encontro	- Interatividade com o sensor <i>Kinect</i>
Demais Encontros	<ul style="list-style-type: none"> <li>- Dedicado aos projetos individuais</li> <li>- Socialização das atividades com os demais colegas</li> </ul>



De forma a complementar a dinâmica do jogo de tabuleiro, faz parte do primeiro encontro a proposta de utilização de jogos educacionais digitais que possuem o propósito de promover a aprendizagem de estruturas de programação, tais como o Estúdio *Code* (VARGAS, 2016). Também é previsto para o primeiro encontro, a exploração do recurso de biblioteca do ambiente *Scratch*, um repositório de aplicações e jogos desenvolvidos e publicados por outros utilizadores. A ferramenta possibilita acessar aos códigos utilizados para o desenvolvimento destas aplicações compartilhadas, permitido ao aluno copiar e adaptar para seu projeto algum recurso que julga interessante, ou ainda através da funcionalidade chamada *Remix* presente no próprio ambiente, copiar integralmente a atividade e adaptá-la para seu propósito.

O segundo e terceiro encontros envolvem os participantes nos primeiros passos utilizando o ambiente *Scratch*. Buscamos neste momento estabelecer relações entre a interface e as atividades experienciadas no jogo de tabuleiro, com o que é apresentado pelo ambiente digital *Scratch*, sendo a transposição dos conhecimentos facilitada nesta proposta pela semelhança na estrutura de construção dos dois ambientes. Para isso, selecionamos um total de seis atividades básicas (MARJI, 2014) que buscam aplicar de forma prática todos os blocos de programação disponíveis na interface do ambiente, além da inserção de personagens e as possibilidades de movimentação, cenários, transições entre telas, uso de comandos básicos (condicionais, movimento, eventos, operadores e variáveis), a inserção de sistemas de pontuação e colisões.

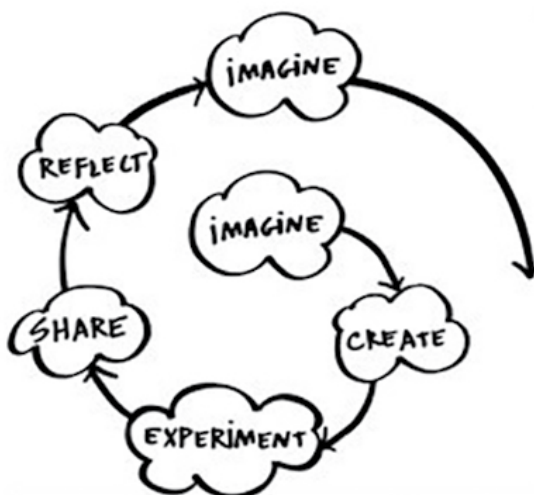
O quarto encontro aborda as características e possibilidades de interação com a interface do jogo a partir do sensor *Kinect*. Para esta etapa, algumas atividades básicas detalham a instalação dos recursos necessários (drivers e adaptador a serem instalados no PC), bem como, o detalhamento dos pontos físicos do corpo humano que podem ser capturados pelo sensor *Kinect* e transferidos para a interface do ambiente *Scratch* por meio de um sistema de coordenadas (“X” e “Y”), além de abordar algumas das técnicas já utilizadas em games comerciais para a interatividade entre o jogador e a tela do produto desenvolvido.

Ao final do quarto encontro, propomos aos participantes o desenvolvimento de um documento de registro das atividades e objetivos propostos, que chamaremos de Game Design Document (MOTTA; TRIGUEIRO JUNIOR, 2013), ou GDD. Neste documento deve constar o objetivo do exergame que o participante pretende desenvolver, os elementos que farão parte da interface, os recursos necessários, e a mecânica de jogo (como será esta experiência para o aluno), levando em consideração que todos estes dados devem estar baseados em um problema, uma questão norteadora. Como recursos de apoio, a oficina disponibiliza aos participantes um site com a descrição de todos os recursos utilizados na oficina, além de links para vídeos tutoriais e materiais de apoio.

Resnick (2007), propõe o ensino do ambiente *Scratch* a partir de uma metodologia em forma de espiral (Figura 4), na qual os utilizadores imaginam o que querem produzir, criam um projeto com base nas suas ideias, experimentam suas criações, partilham suas ideias e criações com outros utilizadores, e reflitam sobre as suas experiências. Isto tudo os levam a imaginar novas ideias e novos projetos:



Figura 4. Método de ensino proposto por Resnick (2007), baseado na teoria do pensamento computacional.



Estes conceitos também irão guiar os procedimentos metodológicos da oficina a partir do sexto encontro, quando iniciam as atividades individuais. Assim, o conceito de “Imaginação”, primeira etapa na construção do produto nesta oficina, deve abarcar não somente a forma e o conteúdo que se pretende trabalhar, mas a forma como este produto deverá ser utilizado pelos alunos. Após criar as primeiras versões, experimentar e compartilhar com os colegas da oficina, o autor do produto irá refletir e voltar a imaginar, em um nível mais alto, com alterações, adaptações, melhoramentos que podem tornar a experiência mais interessante para o seu propósito. Julgamos que o compartilhamento destas informações entre os colegas pode fazer com que reflitam sobre os próprios produtos que estão desenvolvendo, bem como, contribuam com os colegas, apontando melhorias e soluções de forma colaborativa para o resultado da oficina.

### *Considerações finais*

Buscamos delimitar a proposta de uma metodologia de ensino, embasada em conceitos teóricos construtivistas, que seja capaz de proporcionar aos professores de outras áreas do conhecimento, que não ligadas à área de programação ou informática, a possibilidade de se expressar tecnologicamente produzindo um exergame específico, projetado para seu contexto de uso. A proposta justifica-se pela grande quantidade de publicações com resultados positivos acerca da utilização de exergames nas áreas de saúde e educação, mas também, pelo fato de que estas propostas incluam a utilização de recursos comerciais, exigindo uma nem sempre viável adaptação da metodologia ao produto; ou ainda, a utilização de produtos desenvolvidos por profissionais, o que também apresenta dificuldades tais como a complexidade dos conhecimentos específicos necessários ou o alto custo de implementação, que por vezes inviabiliza o projeto.

Para validação da proposta, será realizado posteriormente um programa piloto de intervenção junto à professores de tecnologia educacional de nível fundamental, da prefeitura municipal de Florianópolis, o qual irá compor uma versão final da metodologia.

Será avaliado se as ações relatadas acima são suficientes para engajar os participantes no desenvolvimento de um produto finalizado de sua escolha, e possível de ser aplicado a sua realidade. Neste contexto, o papel do professor que está aplicando a oficina é de “criar um sistema de relações que desafie a atividade do aluno a criar o novo, o não existente ainda” (BECKER, 2012 p.196). No momento em que o participante necessita apropriar-se de conceitos computacionais suficientes, não para reprimir a construção de um dos exemplos discutidos durante a oficina, mas sim, transpondo os conceitos que construiu, desenvolvendo um produto novo, comprovaremos o êxito da proposta. Dado a complexidade em se propor e construir este novo produto, somente uma postura ativa do participante da oficina, será capaz de indicar as ações necessárias, os problemas a serem resolvidos, e a buscar soluções por meio de uma ação consciente.

Por fim, indicamos que estudos futuros sobre aplicabilidade desta metodologia possam ser realizados com profissionais de diferentes áreas, tais como matemática, português, letramento, fisioterapia e educação física.

## Referências

- ANDERSON, F.; ANNETT, M.; BISCHOF, W. F. Lean on Wii: physical rehabilitation with virtual reality Wii peripherals. **Studies in Health Technology and Informatics**, v. 154, n. 154, p. 229-34, 2010.
- ANDRADE, M.; SILVA, C.; OLIVEIRA, T. Desenvolvendo games e aprendendo matemática utilizando o Scratch. In: **Proceedings of the XII SBGames**, 16-18 October 2013, São Paulo, p. 260-263, 2013.
- BECKER, F. **O caminho da aprendizagem em Jean Piaget e Paulo Freire: da ação à operação**. 2. ed. Petrópolis, RJ: Vozes, 2011.
- BECKER, F. **Epistemologia do professor de matemática**. Petrópolis, RJ: Ed. Vozes, 2012.
- BECKER, F. **A Epistemologia do Professor: o cotidiano da escola**, Vozes, 2003.
- CHAPMAN, E. Alternative approaches to assessing student engagement rates. **Practical Assessment, Research & Evaluation Journal**, v. 13, n. 8, p. 251-263, 2003.
- FINCO, M. ; REATEGUI, E.; ZARO, M. A.; SHEEHAN, D.; KATZ, L. Exergaming as an Alternative for Students Unmotivated to Participate in Regular Physical Education Classes. **International Journal of Game-Based Learning**, v. 5, p. 1-10, 2015.
- HOWELL, S. **Kinect2Scratch (Version 2.5)**, [Computer Software]. 2012.
- MALONEY, J. et al. **The scratch programming language and environment**. ACM Transactions on Computing Education (TOCE), v. 10, n. 4, p. 16. 2010.
- MARJI, M. **Aprendendo a programar com Scratch**. São Paulo: Novatec Editora Ltda, 2014.
- MARQUES, T. B. I. Epistemologia Genética. In: SARMENTO, D. F.; RAPOPORT, A.; FOSSATTI, P. **Psicologia e Educação: Perspectivas teóricas e implicações educacionais**, Editora Salles, Canoas, 2008.
- MÉLO, F. É. N.; CUNHA, R. R. M.; SCOLARO, D. R.; CAMPOS, J. L. Do Scratch ao Arduino: Uma proposta para o ensino introdutório de programação para cursos superiores de tecnologia. In: **XXXIX Congresso Brasileiro de Educação em Engenharia**, 2011. Acesso em: <http://198.136.59.239/~abengeorg/CobengeAnteriores/2011/sextoestec/art1886.pdf>
- MOTTA, R. L.; TRIGUEIRO JUNIOR, J. Short game design document (SGDD). In: **Proceedings of the XII SBGames**, 16-18 October 2013, São Paulo, p. 7, 2013. Acessado em [http://www.sbgames.org/sbgames2013/proceedings/artedesign/15-dt-paper\\_SGDD.pdf](http://www.sbgames.org/sbgames2013/proceedings/artedesign/15-dt-paper_SGDD.pdf).

OH, Y.; YANG, S. Defining exergames & exergaming. **Proceedings of Meaningful Play**, p. 1-17, 2010. Acesso em: [https://www.researchgate.net/publication/230794344\\_Defining\\_exergames\\_exergaming](https://www.researchgate.net/publication/230794344_Defining_exergames_exergaming)

PIAGET, J.; GRECO, P. **Aprendizagem e conhecimento**. Rio de Janeiro: Freitas Bastos, 1974.

PIAGET, J. **Abstração reflexionante** – Relações lógico-aritméticas e ordem das relações espaciais. Porto Alegre RS: Artes médicas, 1977.

RESNICK, M. All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten. In: **Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition**. ACM, p. 1-6, 2007.

SCAICO, P. D.; LIMA, A. A.; SILVA, J. B. B.; AZEVEDO, S.; PAIVA, L. F.; RAPOSO, E. H. S.; ALENCAR, Y.; MENDES, J. P. Programação no ensino médio: uma abordagem de ensino orientado ao design com Scratch. In: **Anais do Workshop de Informática na Escola**, v. 1, n. 1, 2012.

SILVA, A. M. S.; MONTANÉ, F. A. T. O uso do Scratch para produção textual no processo de ensino e aprendizagem. In: **Anais do Encontro Virtual de Documentação em Software Livre e Congresso Internacional de Linguagem e Tecnologia Online**, v. 1, n. 5, 2016. Acesso em: [http://www.periodicos.letras.ufmg.br/index.php/anais\\_linguagem\\_tecnologia/article/view/10493](http://www.periodicos.letras.ufmg.br/index.php/anais_linguagem_tecnologia/article/view/10493)

VARGAS, R. M. Ambiente de programação scratch voltada para crianças sob a perspectiva da engenharia semiótica. **Projetos e Dissertações em Sistemas de Informação e Gestão do Conhecimento**, v. 5, n. 1, 2016.

## ***Sobre autores***

### **Luciano Kercher Greis**

Professor de Tecnologias Educacionais no ensino fundamental da rede municipal de Florianópolis – SC. Doutorando em Educação na Universidade do Estado de Santa Catarina, na linha Educação, Comunicação e Tecnologia. Integrante do Laboratório de Gênero, Educação, Sexualidade e Corporeidade (LAGESC) na UDESC na linha de pesquisa Educação, Comunicação e Tecnologia.

### **Kamyla Thais Dias de Freitas**

Doutoranda em Educação na Universidade do Estado de Santa Catarina, na linha Educação, Comunicação e Tecnologia. Integrante do Laboratório de Gênero, Educação, Sexualidade e Corporeidade (LAGESC) na UDESC na linha de pesquisa Educação, Comunicação e Tecnologia.

### **Fernando Luiz Cardoso**

Doutor em Sexualidade Humana - Institute for Advanced Study in Human Sexuality. Professor Titular junto ao Programa de Pós-Graduação em Ciência do Movimento Humano e do Programa de Pós-Graduação em Educação da UDESC. Coordenador do Laboratório de Gênero, Educação, Sexualidade e Corporeidade (LAGESC). Líder do Grupo de Pesquisa Saúde, Educação, Movimento & Evolução (CNPq).



