

DÊ UMA CHANCE AOS Ps: PROJETOS, PARCERIAS, PAIXÃO, PENSAR BRINCANDO

Mitchel Resnick

Resumo

Para prosperar no mundo de hoje, que está em constante mudança, os jovens devem aprender a pensar e agir de maneira criativa. Este documento discute como nosso grupo Lifelong Kindergarten, do MIT Media Lab, desenvolveu a linguagem de programação e a comunidade on-line Scratch com o claro objetivo de ajudar os jovens a se desenvolverem como pensadores criativos. O documento analisa a forma como o Scratch foi criado a partir da perspectiva dos Quatro Ps da Aprendizagem Criativa: Projetos, Parcerias, Paixão, e Pensar brincando.

Palavras-chave: Scratch, codificação, programação, construcionismo, criatividade

1. Introdução

Vivemos em um mundo que muda em uma velocidade cada vez mais impressionante. As crianças de hoje enfrentarão um fluxo contínuo de novos problemas e desafios inesperados no futuro. Muito do que elas aprendem hoje cairá em desuso amanhã. Para serem bem-sucedidas, elas devem aprender a desenvolver soluções inovadoras para os problemas inesperados que, sem dúvida, surgirão em suas vidas. Seu sucesso e satisfação terão como base a capacidade de pensar e agir de maneira criativa. O conhecimento, por si só, não é suficiente: elas precisam aprender a usar esse conhecimento com criatividade.

Como podemos ajudar os jovens de hoje a crescerem como pensadores criativos? Meu grupo de pesquisa, o Lifelong Kindergarten (<http://llk.media.mit.edu>), do MIT Media Lab, tem desenvolvido novas tecnologias, atividades e estratégias para envolver os jovens em experiências de aprendizagem criativa, para que eles possam se desenvolver como pensadores criativos. Nossa abordagem é baseada em quatro elementos fundamentais, que muitas vezes chamamos de "Quatro Ps da Aprendizagem Criativa":

- *Projetos.* Aprendemos melhor quando trabalhamos ativamente em projetos significativos, criando novas ideias, desenvolvendo protótipos e refinando o trabalho por meio da repetição.
- *Parcerias.* O aprendizado prospera quando é feito como uma atividade social, com pessoas compartilhando ideias, colaborando em projetos e ajudando no trabalho umas das outras.

- *Paixão*. Quando as pessoas trabalham em projetos pelos quais têm interesse, elas trabalham por mais tempo e se esforçam mais, persistem diante dos desafios, e aprendem mais nesse processo.
- *Pensar brincando*. Aprender envolve experiências divertidas, ou seja, testar coisas novas, manipular diferentes materiais, testar limites, assumir riscos, repetir algo várias vezes.

Esses quatro Ps estão estritamente alinhados à abordagem do Construcionismo para a educação (e inspirados nela), que enfatiza o valor da criação de projetos que sejam significativos para os alunos, de maneira divertida e em colaboração com colegas [2] [5].

Este documento descreve como o grupo Lifelong Kindergarten usou os Quatro Ps da Aprendizagem Criativa como princípios para o projeto da linguagem de programação e comunidade on-line Scratch (<http://scratch.mit.edu>). Desde o lançamento do Scratch, em 2007, milhões de jovens de todo o mundo usaram o Scratch para programar suas próprias histórias interativas, jogos e animações, e também para compartilhar essas criações uns com os outros on-line [7] [8]. O Scratch teve forte inspiração na linguagem de programação Logo [4], mas vai além dela, tornando a programação *mais manipulável* (oferecendo uma abordagem gráfica de blocos à programação – Fig. 1), *mais significativa* (dando suporte a vários tipos de projetos, para que pessoas com uma ampla gama de interesses possam trabalhar em projetos pelos quais tenham interesse), e *mais social* (permitindo aos alunos compartilhar, recriar a partir do trabalho do outro, comentar e colaborar com os projetos uns dos outros).

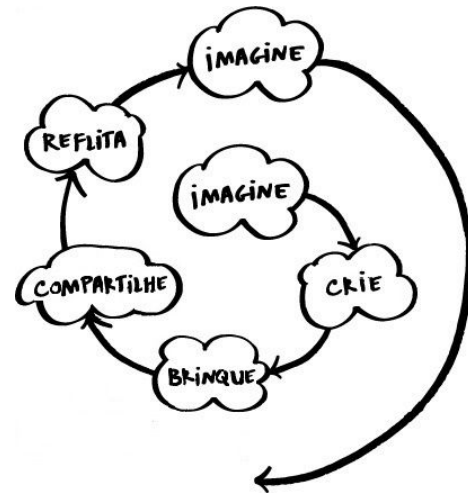


Fig. 1: Editor de programação Scratch

As seções a seguir descrevem como o Scratch (e as atividades relacionadas a ele) foi orientado pelos Quatro Ps da Aprendizagem Criativa: Projetos, Parcerias, Paixão, e Pensar brincando.

2. Projetos

O Scratch foi desenvolvido com base em projetos. Já foram criados e compartilhados mais de 5 milhões de projetos na comunidade on-line Scratch, com cerca de 10 mil novos projetos por dia. Quando os jovens criam projetos do Scratch, eles se envolvem na “espiral do pensamento criativo” [6]: eles *imaginam* o que querem fazer, *criam* um projeto com base em suas ideias, *brincam* com suas criações, *compartilham* suas ideias e criações com outras pessoas, *refletem* sobre suas experiências – e tudo isso os leva a *imaginar* novas ideias e projetos. Enquanto os alunos passam por esse processo, cada vez mais eles aprendem a desenvolver suas próprias ideias, testá-las, testar limites, experimentar alternativas, ouvir as opiniões dos outros e criar novas ideias com base em suas experiências. Nesse processo, eles se desenvolvem como pensadores criativos.



Para nós, parece natural apresentar a programação aos jovens por meio de projetos, para que eles aprendam a se expressar criativamente à medida que aprendem a programar. No entanto, vários programas de introdução à programação adotam uma abordagem bastante diferente. Pense, por exemplo, nos tutoriais de introdução da Hora do Código, uma campanha de grande destaque para levar a programação aos jovens (<http://csedweek.org>). O tutorial do Scratch para a Hora do Código (criado por Karen Brennan e colegas) foi explicitamente baseado em projetos, ajudando os jovens a criar seus próprios cartões interativos de boas-festas. Milhares de jovens do mundo todo programaram esses cartões e compartilharam suas criações no site do Scratch (Fig. 3).

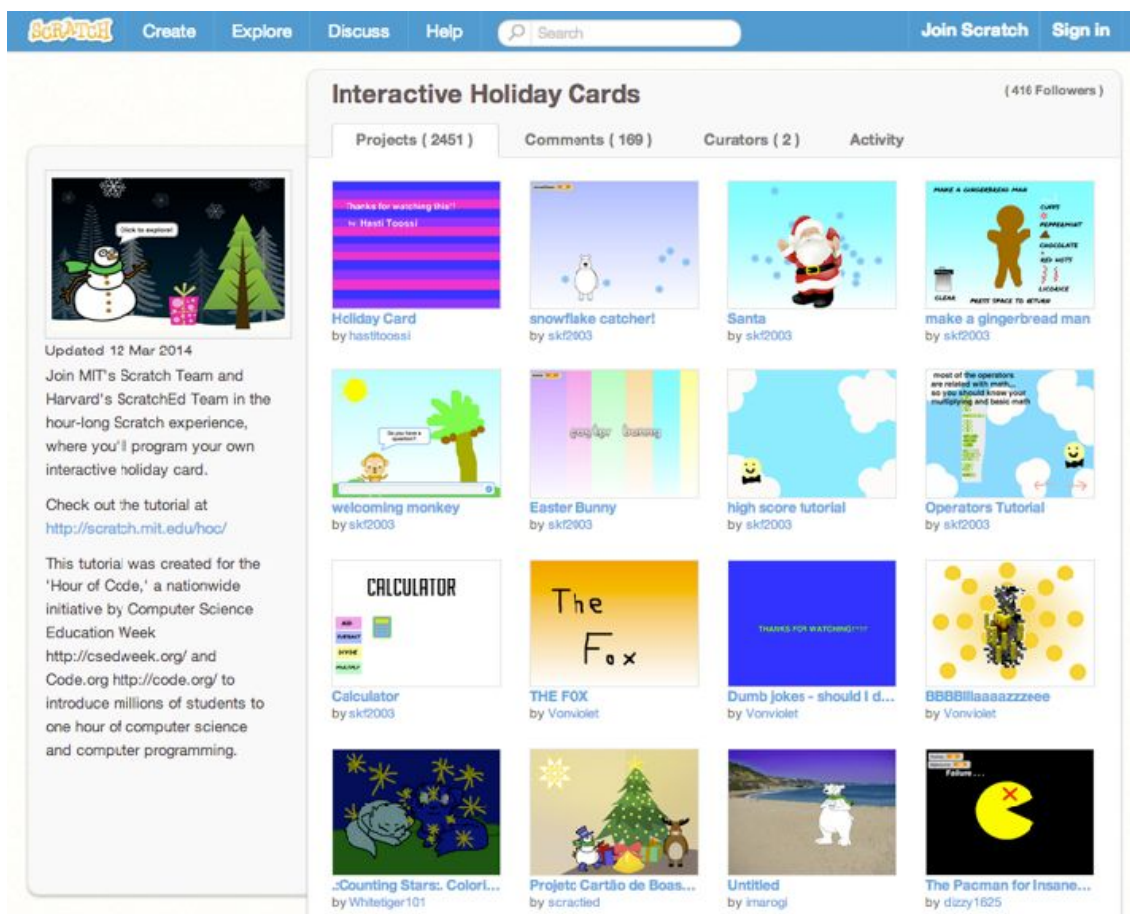


Fig. 3: Projetos do Scratch para a Hora do Código

Por outro lado, a maioria dos outros tutoriais da Hora do Código tem o formato de desafio, e não de projeto. Esses tutoriais apresentaram aos alunos uma série de desafios de lógica, nos quais eles precisam programar personagens animados para movê-los de um lugar a outro (Fig. 4). Quando os alunos resolvem um desafio, eles podem passar para o próximo. Sem dúvidas, os alunos aprendem a usar alguns conceitos computacionais úteis ao trabalhar nesses desafios, mas aprender a programar resolvendo desafios de lógica é como aprender a escrever resolvendo desafios de palavras cruzadas. Não é assim que nos tornamos verdadeiramente fluentes. Assim como os alunos que estão desenvolvendo a fluência em um idioma precisam de oportunidades para escrever histórias (e não apenas brincar com jogos de palavras), os alunos que estão desenvolvendo a fluência em programação precisam de oportunidades de criar projetos (e não só resolver desafios).

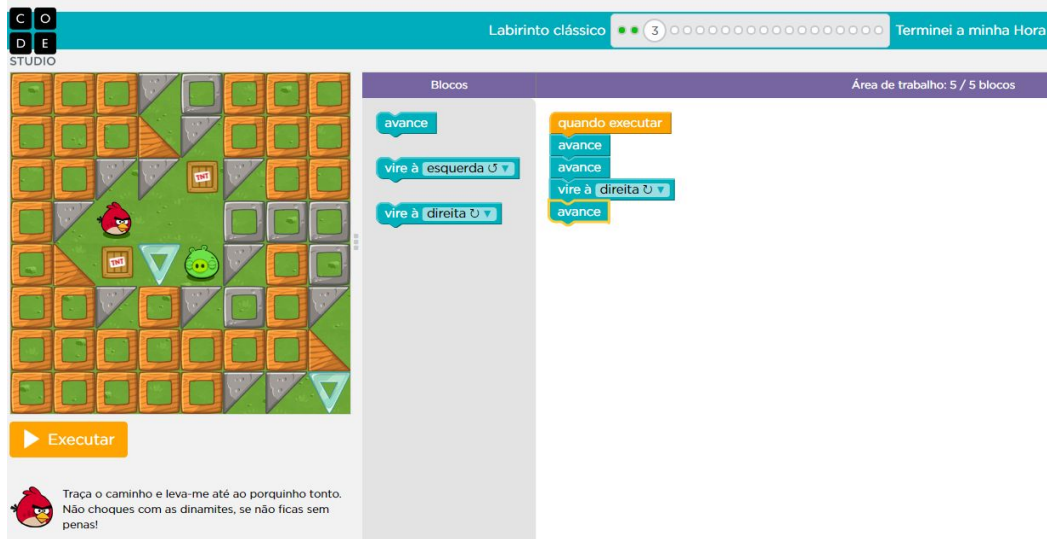


Fig. 4: Modelo de tutorial de desafio da Hora do Código

3. Parcerias

Quando lançamos a linguagem de programação Scratch, em 2007, lançamos a comunidade on-line Scratch ao mesmo tempo. Acreditamos, desde o princípio, que a interação com colegas deveria ser um elemento central no processo de aprendizado.

A comunidade on-line Scratch tem dois papéis: fornecer um *público* (quando os jovens compartilham seus projetos, eles recebem *feedbacks* e sugestões de colegas da comunidade), e, ao mesmo tempo, oferecer *inspiração* (quando os jovens testam projetos feitos por seus colegas, eles podem receber novas ideias, e até mesmo pegar *scripts* e imagens emprestados) para seus próprios projetos [1].

Desenvolvemos o Scratch de modo que as pessoas achassem fácil *criar a partir de outros projetos (remix)* [3]. Todos os projetos na comunidade Scratch têm licença Creative Commons, assim, é possível se basear no trabalho e nas ideias de outras pessoas, contanto que o crédito seja atribuído de maneira adequada. Mais de um quarto dos projetos no site do Scratch são recriações a partir de outros projetos. O site oferece visualizações para mostrar como as ideias se espalham entre os colegas na comunidade (Fig. 5).

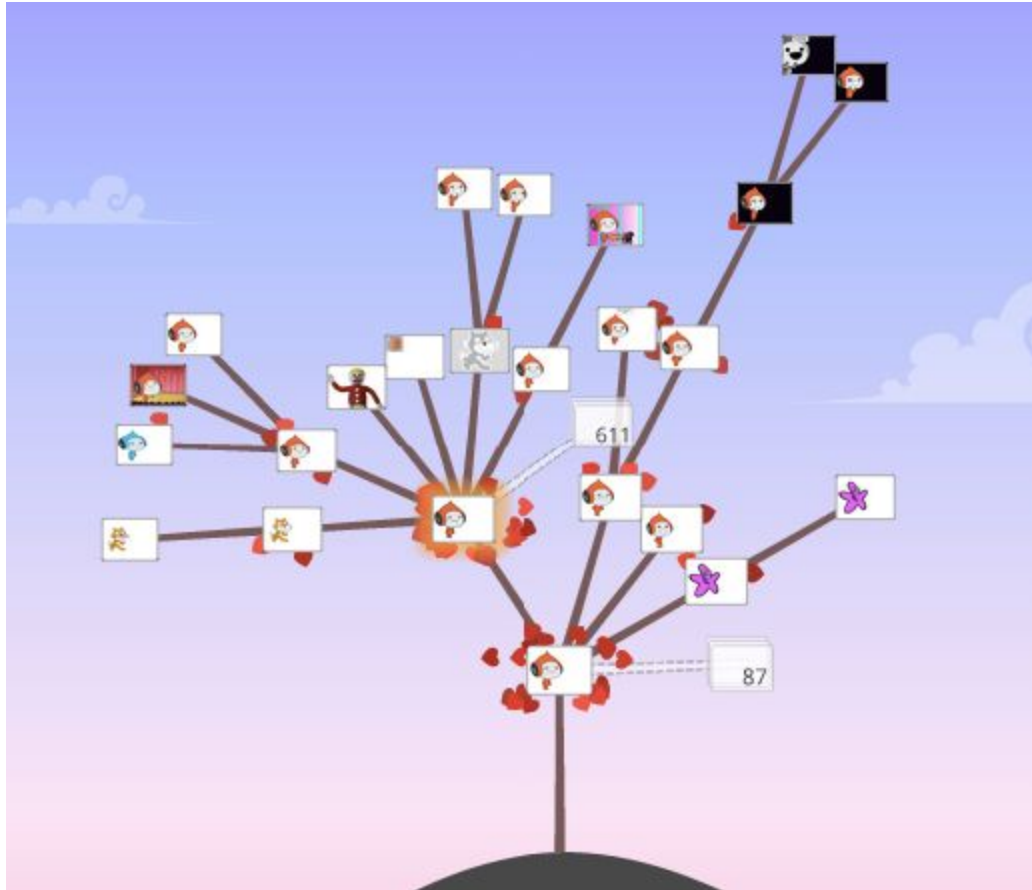


Fig. 5: Árvore de recriações

Ficamos surpresos ao ver o quanto os membros da comunidade Scratch estão dispostos a ajudar uns aos outros. Quando criamos o Scratch, esperávamos que alguns professores criassem tutoriais, mas não que os jovens o fizessem. Na verdade, os jovens criaram e compartilharam milhares de tutoriais, ajudando outras pessoas a aprender a programar, a desenhar personagens gráficos e inclusive a aumentar a popularidade de seus projetos no site do Scratch (Fig. 6). Os jovens Scratchers também criaram galerias onde podem pedir (e oferecer) ajuda para projetos. É esse tipo de compartilhamento e colaboração que faz do Scratch uma comunidade de aprendizagem criativa.

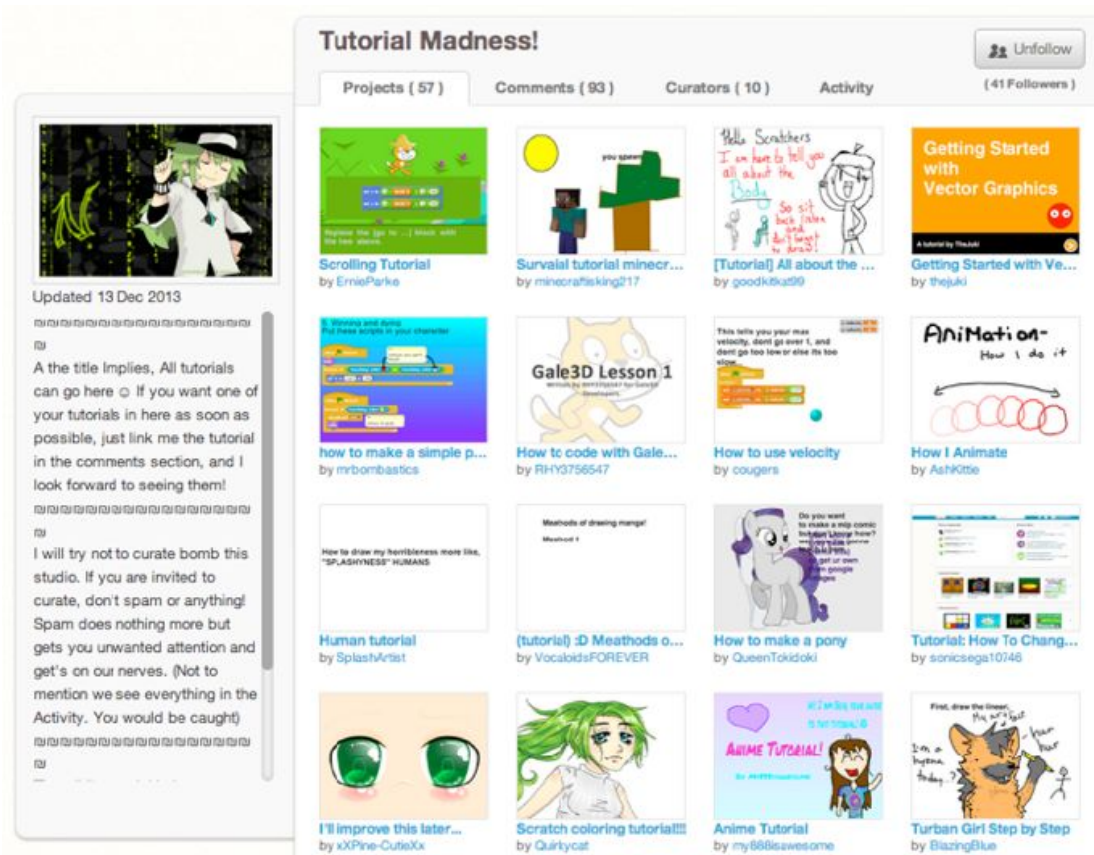


Fig. 6: Tutoriais de Scratchers para Scratchers

4. Paixão

É uma grande alegria ver que os jovens compartilharam mais de 5 milhões de projetos no site do Scratch. Mas estamos ainda mais felizes em ver a *diversidade* dos projetos no site. Eles não só compartilharam histórias e jogos, mas também cartões de aniversário interativos, tirinhas de animes, boletins informativos interativos, *tours* virtuais, concursos de dança, anúncios de serviço público, e muito mais (Fig. 7).

Por que a diversidade é importante? Ela é um indicativo de que os jovens estão trabalhando em projetos do Scratch com os quais eles realmente se importam. Essa diversidade dos projetos é um reflexo da diversidade de interesses dos jovens. Com o Scratch, jovens com diferentes experiências e interesses podem trabalhar nos projetos de que gostam. E quando as pessoas trabalham com o que gostam, elas tendem a se esforçar e aprender mais.



Fig. 7: Variedade de projetos do Scratch

Quando criamos o Scratch, enfatizamos o uso de mídias (músicas, sons, imagens, fotos), pois sabemos que muitos jovens são apaixonados por elas. Alguns pais e educadores mostram-se céticos sobre o papel central das mídias nos projetos do Scratch, pois se preocupam que elas possam distrair os jovens dos aspectos mais "educativos". Veja esta mensagem que recebemos de um dos pais que, inicialmente, se mostrava cético, mas que começou a valorizar a função desse elemento no Scratch:

Eu tenho que admitir que, no início, não entendia por que uma linguagem de programação para crianças precisava ser tão focada nas mídias, mas depois de ver meus filhos interagindo com o Scratch, isso ficou claro para mim. Uma das coisas mais legais que vi no Scratch foi que ele personalizou a experiência de desenvolvimento de maneiras novas, facilitando que meus filhos adicionassem conteúdos personalizados e participassem ativamente do processo de desenvolvimento. Eles conseguiram não só desenvolver programas abstratos para fazer coisas simples com um gato ou uma caixa, etc., eles também conseguiram adicionar suas PRÓPRIAS imagens e suas PRÓPRIAS vozes ao ambiente do Scratch, o que rendeu horas de diversão e os levou a aprender.

5. Pensar brincando

Quando a maioria das pessoas pensa sobre o conceito de “brincar”, elas se lembram de diversão e alegria. O meu grupo de pesquisa, no entanto, pensa nisso de uma forma diferente. Para nós, brincar é uma atitude e uma forma de se relacionar com o mundo. Associamos a brincadeira com a possibilidade de assumir riscos, testar coisas novas e testar limites. Vemos o ato de brincar como um processo de manipulação, experimentação e exploração, e esses aspectos são fundamentais para o processo de aprendizagem criativa.

Quando criamos o Scratch, queríamos incentivar os jovens a manipular, experimentar e brincar com seus projetos. Desenvolvemos os blocos de programação do Scratch para serem manipuláveis. É possível encaixar e separar os blocos facilmente, assim como peças de LEGO. Você pode testar e repetir seus *scripts* de programação continuamente.

Da mesma forma, o site do Scratch incentiva os jovens a manipular e testar projetos. É possível arrastar *scripts*, imagens e sons de um projeto para sua "mochila" e, então, movê-los para um outro projeto. Também é possível marcar um projeto como "rascunho", assim as outras pessoas saberão que o trabalho está em andamento.

Por exemplo, um Scratcher com o nome de usuário EmeraldDragon começou a trabalhar em um jogo com um dragão que corre pela tela (Fig. 8a). Nas anotações, EmeraldDragon escreveu: "Só fiz isso por enquanto. Estou trabalhando para conseguir correr para frente e para trás sem que a pedra desapareça. Alguém tem dicas ou pode ajudar? Essa é só uma fase de um longo processo". EmeraldDragon nomeou o projeto como *Meu jogo de dragão (NÃO concluído)*.

Pouco tempo depois, EmeraldDragon adicionou um comentário ao projeto: "Eu estava manipulando os *scripts* no jogo e finalmente descobri como fazer para correr para frente e para trás! Vou corrigir o jogo e disponibilizar a versão nova e melhorada dele, mas ela ainda não está concluída".

Depois disso, EmeraldDragon publicou uma versão aprimorada do jogo, incluindo um novo processo para controlar a pontuação (Fig. 8b). Mas EmeraldDragon queria enfatizar que o projeto ainda era um trabalho em andamento. Título do projeto revisado: *Meu jogo de dragão (Ainda NÃO concluído)*.



Fig. 8a



Fig. 8b

6. Só estamos dizendo para você dar uma chance aos Ps

Nos últimos anos, houve um grande aumento no interesse em inventar e programar. Espaços de invenção e clubes de programação estão sendo abertos em todos os lugares. O entusiasmo em torno do Movimento "maker" (inventor) e do Movimento do programador traz a oportunidade de revigorar e revalidar a tradição do Construcionismo na educação. Mas, inventar e programar não é o suficiente. Para ajudar os jovens a se prepararem para um mundo que se modifica mais rapidamente do que nunca, precisamos incorporar invenção e programação em um processo de aprendizagem criativa caracterizado por Projetos, Parcerias, Paixão e Pensar brincando.

Agradecimentos

Vários membros do grupo Lifelong Kindergarten do MIT Media Lab contribuíram para as ideias e tecnologias descritas neste documento. Em particular, a estrutura dos Quatro Ps da Aprendizagem Criativa foi desenvolvida em estreita colaboração com Natalie Rusk e Philipp Schmidt, como parte de nosso curso on-line "Learning Creative Learning" (<http://learn.media.mit.edu>).

Referências

- [1] K. Brennan, M. Resnick, and A. Monroy-Hernandez, "Making projects, making friends: Online community as catalyst for interactive media creation," in *New Directions for Youth Development*, vol. 128, 2010.
- [2] Y. Kafai and M. Resnick, *Constructionism in practice: Designing, thinking, and learning in a digital world*. Lawrence Erlbaum, 1996.
- [3] A. Monroy-Hernandez, *Designing for Remixing: Supporting an Online Community of Amateur Creators*. PhD dissertation, MIT Media Lab, 2012.
- [4] S. Papert, *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, 1980.

- [5] S. Papert, "Situating Constructionism," in *Constructionism*, I. Harel and S. Papert, Eds. Ablex Publishing, 1991.
- [6] M. Resnick, "All I Really Need to Know (About Creative Thinking) I Learned (By Studying How Children Learn) in Kindergarten," in *ACM Creativity & Cognition conference*, 2007.
- [7] M. Resnick, J. Maloney, A. Monroy-Hernandez, N. Rusk, E. Eastmond, K. Brennan, A. Millner, E. Rosenbaum, J. Silver, B. Silverman, and Y. Kafai, "Scratch: Programming for All," in *Communications of the ACM*, vol. 52, no. 11, 2009, pp. 60-67.
- [8] M. Resnick, "Reviving Papert's Dream," in *Educational Technology*, vol. 52, no. 4, 2012, pp. 42-46.