

Capítulo

6

Programando, Criando e Inovando com o Scratch: em busca da formação do cidadão do século XXI

Elaine Silva Rocha Sobreira, Olga Kikue Takinami e Verônica Gomes dos Santos

Abstract

This chapter proposes a reflection and discussion of possibilities of using the Scratch programming language, for professionals who wish to know and work this language in Elementary Education, with the aim of developing skills and abilities necessary for the new profile of active citizen in the XXI century. The syllabus aims to address reflexively contributions of working with the programming language for the development of collaborative learning, interactive, functional, constructive and meaningful, focusing on the role and autonomy of the student, beyond make suggestions for development projects and provide Scratch's interaction with other interfaces and resources that enable access for people with disabilities.

Resumo

Este capítulo propõe a reflexão e a discussão de possibilidades de utilização da linguagem de programação Scratch, por profissionais que desejam conhecer e trabalhar esta linguagem no Ensino Fundamental, com o intuito de desenvolver competências e habilidades necessárias para o novo perfil de cidadão atuante no século XXI. O conteúdo programático visa abordar de forma reflexiva as contribuições do trabalho com a linguagem de programação para o desenvolvimento de uma aprendizagem colaborativa, interativa, funcional, construtiva e significativa, com foco no protagonismo e autonomia do aluno, além de apresentar sugestões de desenvolvimento de projetos, interação do Scratch com outras interfaces e recursos que viabilizem o acesso de pessoas com deficiência.

6.1. Introdução

O Scratch é uma linguagem de programação educacional, criada a partir da linguagem Logo (desenvolvida por Seymour Papert). Foi desenvolvida pela equipe do *Lifelong Kindergarten*, coordenada por Mitchel Resnick do MIT - *Massachusetts Institute of Technology*, em 2007. Tem código aberto e é gratuito, além de permitir uma fácil instalação.

Por ser intuitivo e permitir fácil programação através de seus blocos de encaixe, torna-se ideal para o trabalho com crianças a partir de oito anos de idade, pois dispensa qualquer conhecimento prévio de linguagem de programação. Com o Scratch é possível criar jogos, animações, histórias animadas, entre outros recursos que serão descritos ao longo do texto, além das possibilidades de comunicação com outras interfaces e de associação com recursos de acessibilidade.

O trabalho com programação desde a infância vem atender a mudança no perfil da sociedade atual, considerada “sociedade midiática”, a qual requer também um novo perfil de cidadão, que possa interagir, transformar, protagonizar e atuar diante dos recursos tecnológicos de forma consciente e ativa. Já não basta o simples consumo ou manuseio mecânico dos produtos disponíveis no mercado: esta sociedade necessita de indivíduos que sejam capazes de se deparar com a diversidade de situações do cotidiano e possam agir sobre elas, solucioná-las, transformá-las. Para tanto, há de se desenvolver competências e habilidades específicas para esta atuação.

Compreendendo que tais habilidades podem, dentre outras formas, ser desenvolvidas através do contato e do trabalho com linguagens de programação desde o Ensino Fundamental, este texto busca oferecer oportunidades para o educador refletir sobre as possibilidades de criação e de programação com o Scratch, integrando-o num currículo escolar que busca desenvolver competências próprias do cidadão do século XXI, através da vivência e de discussões aliadas à socialização de práticas pedagógicas que se aplicam, também, aos princípios que pautam a educação inclusiva, isto é, a diversidade.

Desta forma, propõe-se primeiramente a discussão sobre a transformação social como resultado do avanço tecnológico e, conseqüentemente, a necessidade de formação de um novo perfil de cidadão, o qual pode ser potencializado através de uma educação que incentive a reflexão e a discussão no decorrer do trabalho com linguagem de programação.

No tocante à pertinência pedagógica e à elaboração de propostas contextualizadas e desafiadoras, objetiva-se refletir sobre o uso da programação para o desenvolvimento de competências de aprendizagens integrando todos os alunos na sua diversidade, além de buscar um aprofundamento e reflexão acerca das propostas e como podem ser desafiadoras para todos, considerando os estilos de aprendizagem.

Assim, a estrutura deste texto está organizada em cinco itens. Inicia-se com embasamento teórico fundamentada em autores renomados, em que discute a importância de trabalhar a programação numa perspectiva construcionista. Em seguida aborda-se a sociedade do século XXI, em como deve ser a formação do aluno e como propor desafios que atendam a todos. No terceiro é apresentado o Scratch seguido de uma discussão do seu uso para o desenvolvimento de habilidades e algumas possibilidades pedagógicas com exemplos de práticas, apresentamos também o uso de outras interfaces que se comunicam com o Scratch. No quarto item, os recursos de acessibilidade que podem auxiliar os alunos com deficiência. Por fim, as considerações finais que delineiam as conclusões das autoras.

6.2. Embasamento Teórico

Dois aspectos merecem muita atenção em nossas escolas. O primeiro é o protagonismo e a autoria dos alunos quando se utiliza as TDIC (Tecnologias Digitais de Informação e Comunicação) e o segundo, igualmente importante, é a homogeneidade das atividades propostas para um público tão heterogêneo que a escola passou a receber.

Nossos alunos apresentam grandes facilidades para se familiarizar com as tecnologias. A interação das crianças e dos jovens com as tecnologias, de acordo com Prensky (2001), acarretou em uma modificação no modo de pensar e processar as informações nas gerações atuais. Esta geração é denominada por Prensky de “nativos digitais”, pois são nativos na linguagem digital, ou seja, o contato com as diferentes tecnologias é intrínseco ao seu crescimento. Estes nativos digitais possuem fluência no envio de mensagens, na manipulação de jogos, arquivos, músicas, vídeos, filmes e *softwares* diversos, mas muitos, não sabem como criar *softwares* ou jogos com o computador, sabem apenas manipular o que foi programado por outras pessoas, conforme alerta Mark Surman (2013), diretor-executivo da Mozilla Foundation, para a necessidade de promover uma transformação na atual geração: o de consumidora para o de criadora de tecnologia. Sendo assim, precisamos buscar uma fluência digital, que requer não apenas a capacidade de comunicar, navegar e interagir na *web*, mas também a capacidade de criar e inventar com as novas mídias.

O trabalho com a linguagem de programação tem sua importância na educação pelo fato de propiciar o desenvolvimento da criatividade e uma nova forma de expressar ideias. Para Souza (2013) o domínio desta linguagem possibilita, também que as pessoas se expressem no ambiente computacional e *online*, além de desenvolver o espírito de colaboração e disseminar o compartilhamento de ideias, fundamental para se ter uma participação social plena. Atualmente, há uma crescente valorização desta prática nas séries iniciais, de modo a garantir que o aluno não seja apenas um mero consumidor de informações, passivo, principalmente após a chegada dos *tablets*, mas que seja capaz de produzir seus próprios conteúdos, jogos, entre outros recursos, descobrindo como criar e atuar em um ambiente informatizado.

Por ser uma linguagem de programação que permite ao aluno representar simbolicamente o raciocínio lógico e desenvolver estratégias, o Scratch possibilita o uso por alunos com deficiência, pois oferece um cenário para o desenvolvimento de diferentes inteligências, como vemos abaixo na Figura 6.1. Seus blocos de encaixe (movimento, aparência, som, caneta, controles, sensores, operadores e variáveis) tornam possível que o aluno perceba a sua ação, observe seus resultados e detalhes concretamente.

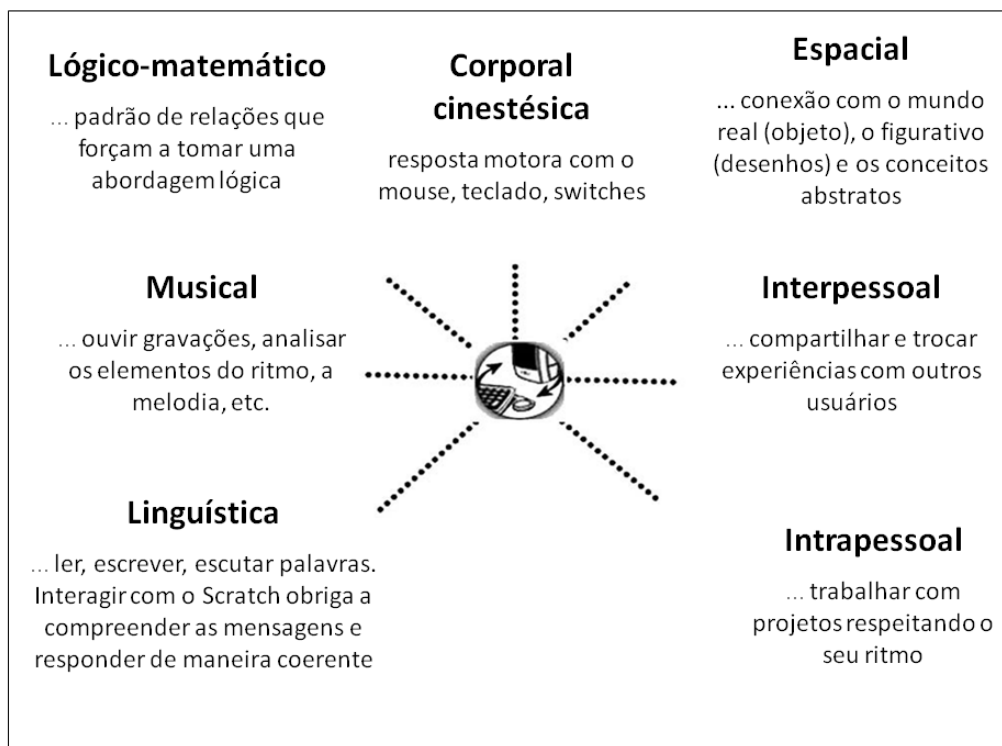


Figura 6.1. Scratch: desenvolvimento de diferentes inteligências¹

Pelo fato de o Scratch ser atraente e de fácil manuseio para o público infantil (a partir de oito anos de idade), o desenvolvimento da programação torna-se possível nesta faixa etária, além de divertida e significativa. Pode-se aliar o trabalho com a linguagem de programação no Scratch ao currículo escolar do Ensino Fundamental, trabalhando em uma perspectiva que busque desenvolver habilidades próprias para o cidadão do século XXI, o qual cria, gerencia uma diversidade de mídias, desenvolve seu raciocínio lógico na experimentação e resolução de problemas, além de compartilhar seus conhecimentos. Considerando ainda, a diversidade dos alunos em suas condições física, biológica e social com propostas que podem ser apresentadas para que a aprendizagem ocorra, de maneira significativa para todos.

6.3. A sociedade do século XXI

Um crescente movimento ganha corpo na mídia mundial, mobilizando importantes personalidades do universo tecnológico e de criação para a *web*. Tal mobilização faz surgir um novo termo imbricado de significado e significância para o cenário educacional: a “Alfabetização Digital”.

Para muito além do acesso e do consumo de tecnologia e de conteúdo digital, a campanha norte americana encabeçada por nomes como Bill Gates e Mark Zuckerberg, entre outros, propõe uma postura diferenciada perante a tecnologia buscando uma produção e reflexão a partir desta, de modo que seja possível “promover uma transformação na atual geração: de consumidora para criadora de tecnologia” (Surman,

¹ Laptop, andamiaje para La Educación Especial – UNESCO Montevideo. Disponível em: http://www.unesco.org.uy/ci/fileadmin/comunicacion-informacion/Laptop_andamiaje_Edu_Especial.pdf. Acesso em: 22/09/2013.

2013). Para tanto, é preciso proporcionar aos alunos, desde muito cedo, a oportunidade de colocar em jogo situações onde é possível refletir, testar, argumentar, comprovar, refutar hipóteses a partir de projetos significativos e funcionais a sua formação, assumindo uma postura proativa, autoral e dinâmica no processo de ensino e aprendizagem.

Sem dúvida esse novo perfil de aluno transformará a forma de interagir na sociedade e atenderá as necessidades que emergem no século XXI, ou seja, cidadãos conectados, pertencentes a uma rede de construção colaborativa que se encaminha para uma autogestão, onde a comunicação, a informação e a formação extrapolam veículos padronizados e locais formais de educação. Surge, assim, um novo fazer educativo.

Diante deste cenário, nos questionamos: Qual o papel do ensino de programação para a formação deste perfil de aluno, de profissional, de cidadão?

Seymour Papert, desde os anos de 1960, realizou estudos relacionados a interação entre criança, computador e a linguagem Logo, somando-se às ideias de Jean Piaget buscando entender a problemática apresentada à inteligência artificial, propôs com seus estudos mais que uma nova linguagem, uma nova postura filosófica, que surge da relação interacional entre homem-máquina, aluno-computador, surgindo assim o *construcionismo*. Desta reconstrução teórica, Silva (2007) destaca que o trabalho com a linguagem de programação vai muito além de se formar programadores, uma vez que:

Aquilo que a criança aprendeu porque fez, após ter explorado, investigado e descoberto por si própria, além de contribuir para o desenvolvimento de suas estruturas cognitivas, reveste-se de um significado especial que ajuda a reter e transferir com muito mais facilidade aquilo que foi aprendido. Está imbuído na filosofia do Logo, como concebeu Papert, a ideia que a aquisição de um conhecimento não se dá em função do desenvolvimento, mas principalmente na maneira pelas quais as pessoas se relacionam com o meio, ou seja, as condições que este oferece para exercitar o pensamento qualitativo. Acredita na necessidade da pessoa controlar sua aprendizagem, poder reconhecer e escolher entre várias possibilidades de pensamento estruturado. (Silva, 2007, p. 2).

Esta visão sobre o ensino e a aprendizagem apresentada e sustentada por Papert, como vemos na Figura 6.2 e diversos autores considera a aprendizagem por descoberta como a possibilidade de os alunos aceitarem os erros de forma construtiva, fazendo deles uma forma de rever, investigar, entender sua origem e buscar soluções. Esse jogo de vai e vem coloca à prova todo o conhecimento construído pelo aluno, estabelecendo relações com o mundo e sua funcionalidade, acrescentando ao que já se sabe novos níveis de conhecimento para alcançar níveis superiores.

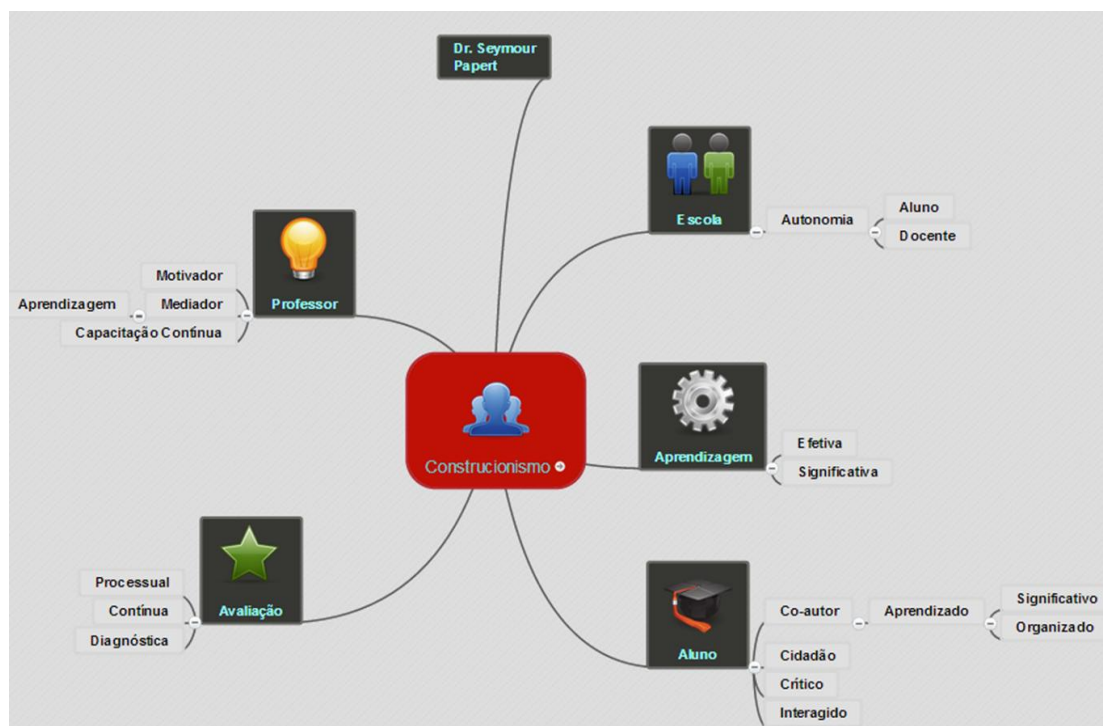


Figura 6.2. Mapa conceitual² sobre o construcionismo de Papert

É possível considerar que o trabalho com a linguagem de programação atua na chamada ZDP (Zona de Desenvolvimento Proximal) sustentada por Vygotsky. Considera-se que esta é:

a distância entre o nível de desenvolvimento real, que se costuma determinar através da solução independente de problemas, e o nível de desenvolvimento potencial, determinado através da solução de problemas sob a orientação de um adulto ou em colaboração com companheiros mais capazes (Vygotsky, 1991, p. 97)

Assim, esta distância entre o que se tem e o que se espera alcançar é um terreno fértil para a promoção de uma aprendizagem significativa que vá além do acúmulo de informações passadas de forma instrucional, linear e unilateral. Para Ausubel (1982) o conhecimento tem que oferecer a possibilidade de ser aplicado além da sala de aula, mas também na vida de forma a tornar-se funcional. Enfim, deve-se despertar o gosto de descobrir, de sentir prazer em saber, em conhecer e de aprender a aprender. Da mesma forma, este espaço de construção, reelaboração, comprovação e validação de conteúdos, conceitos e aprendizagens deve proporcionar a todos os envolvidos o direito de confrontar seus saberes de forma e formato diferenciados, abrangendo a diversidade em todos os seus níveis e possibilidades.

Segundo Felder (2002, *apud* Cavellucci, 2003, p. 8), os estilos de aprendizagem são a maneira como as pessoas processam as informações; habilidades que podem ser desenvolvidas. Há pessoas que necessitam do recurso visual, enquanto outras do auditivo. Uns aprendem quando estão em equipe, outros individualmente. Nesse sentido, o professor não deve contemplar um único estilo de aprendizagem e sim vários, considerando não ser a sala de aula homogênea. Portanto, conhecer, respeitar e

² Mindmeister. Disponível em: http://www.mindmeister.com/export/image/144337631?list_large=true&with_frame=true. Acesso em 22/09/2013.

proporcionar aos alunos o trabalho com variados estilos de aprendizagem é oferecer aos mesmos a oportunidade de atuarem na diferença e na semelhança com a finalidade de que cada aluno construa uma aprendizagem significativa e funcional.

Diante disso, nos perguntamos em que ajudará tal forma de trabalho para a formação do perfil de cidadão para atuar no século XXI. Quem é este novo sujeito e o que se espera dele? Segundo Papert:

Os cidadãos do futuro precisam lidar com desafios, enfrentar um problema inesperado para o qual não há uma explicação preestabelecida. Precisamos adquirir habilidades necessárias para participar da construção do novo ou então nos resignarmos a uma vida de dependência. A verdadeira habilidade competitiva é a habilidade de aprender. Não devemos aprender a dar respostas certas ou erradas, temos de aprender a solucionar problemas. (Papert, *apud* Martins, 2012, p. 18)

Compreendemos assim que o trabalho desenvolvido a partir da linguagem de programação, independente do programa ou base, contribui significativamente para o desenvolvimento de competências e habilidades necessárias para a formação do perfil esperado para atuação no século XXI, envolvendo escolhas, decisões, reformulações, colaboração, interação e, principalmente, criando uma ponte viável entre os saberes considerados estritamente escolares com a vida em sociedade.

Para o desenvolvimento de habilidades possíveis de programação, optamos pela escolha do Scratch, pois é próprio para o trabalho com alunos do Ensino Fundamental. Esta linguagem de programação é apresentada, por Resnick *et al* (2011), na forma de blocos de encaixes, não exige conhecimentos elaborados de programação e propicia o aumento de competências que desejamos, como as de informação e comunicação, raciocínio e resolução de problemas e as interpessoais e de autodirecionamento.

6.4. Sobre o Scratch

Criado para que as pessoas pudessem produzir facilmente suas histórias, jogos, animações e compartilhar uns com os outros, a linguagem de programação Scratch vem sendo muito utilizada na educação básica, em vários locais do mundo, já que propicia uma grande facilidade de programação através de seus blocos que se encaixam (Figura 6.3), não permitindo erros na programação, além de despertar muito o interesse de crianças a partir de oito anos, devido a possibilidade de uma diversidade de criações.

Por ser gratuito e ter código aberto, muitas pessoas investem em tutoriais, sendo possível encontrá-los disponíveis na *web*, permitindo um aprendizado autônomo. Como é um ambiente de autoria, possibilita ao educando a criação do seu próprio objeto de aprendizagem ou criações diversas para o seu lazer, desenvolvendo assim, a criatividade.

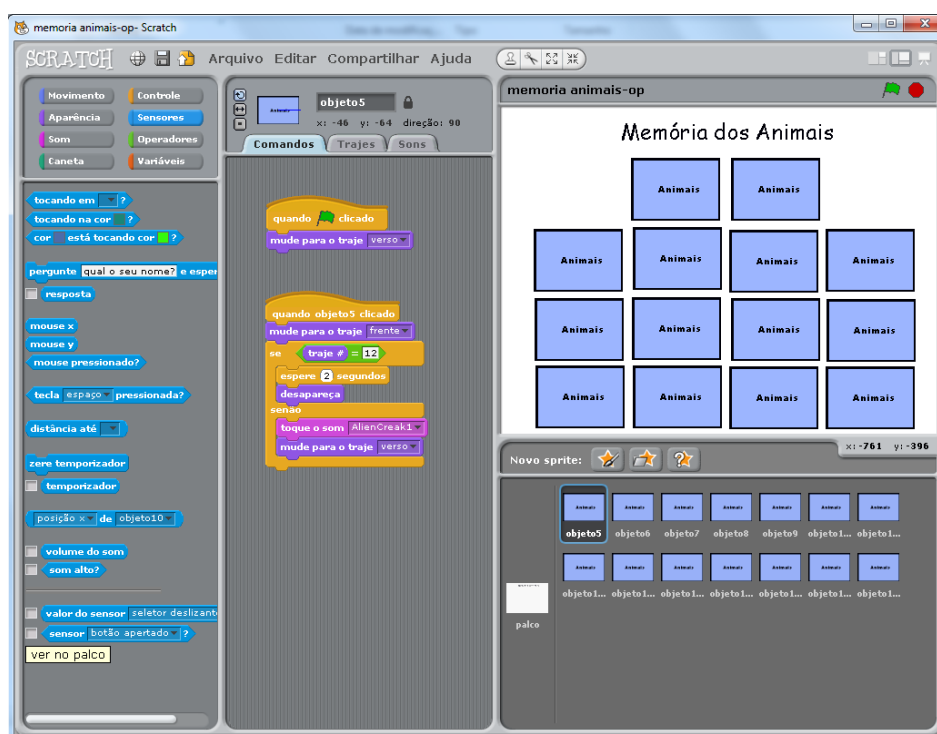


Figura 6.3. Tela do programa Scratch com exemplo de utilização dos blocos

Possui uma interface intuitiva e agradável, propiciando a exploração livre, orientada por tutoriais e criações compartilhadas. Com o Scratch o educando é desafiado a criar animações, simulações, jogos e histórias animadas e compartilhar as produções no *site* oficial do Scratch.mit³.

O *site* oficial (Figura 6.4) é uma comunidade onde ao criar sua conta o usuário compartilha seus projetos, recebem comentários, pesquisa projetos de pessoas de todo o mundo, podendo realizar *downloads*, conhecer o código de criação para, a partir deste, criar novos projetos. Dessa forma, o espírito colaborativo é disseminado nesse ambiente em que os alunos têm a oportunidade de remixar produções diversas criadas por pessoas de todo o mundo. Também é possível editar online, sem baixá-lo, na versão Scratch 2.0, disponível no *site* (Figura 6.5).

³ Scratch – Imagine, Program, Share – MIT. Disponível em: <http://scratch.mit.edu/>. Acesso em: 22/08/2013.

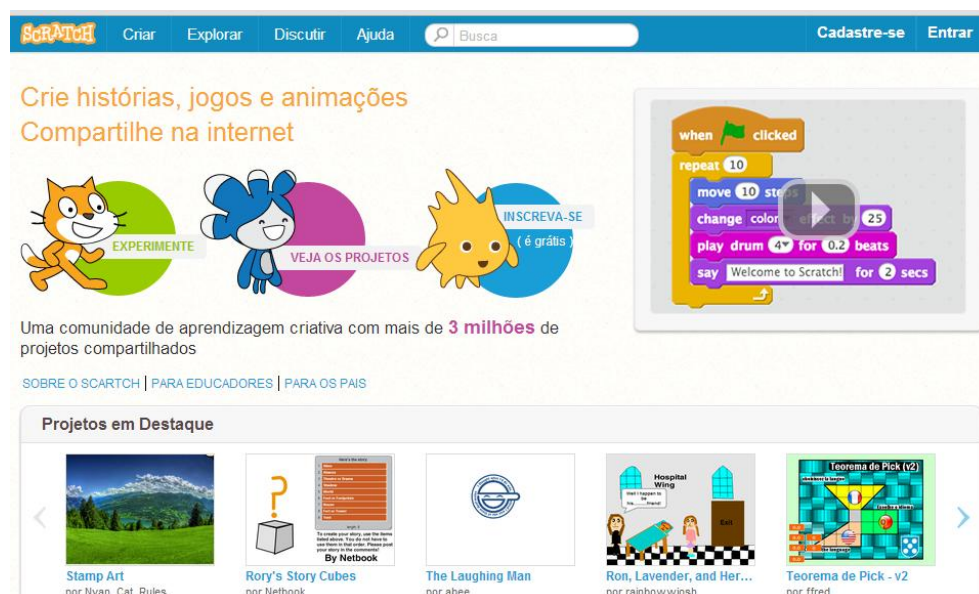


Figura 6.4. Tela de abertura do site Scratch.MIT

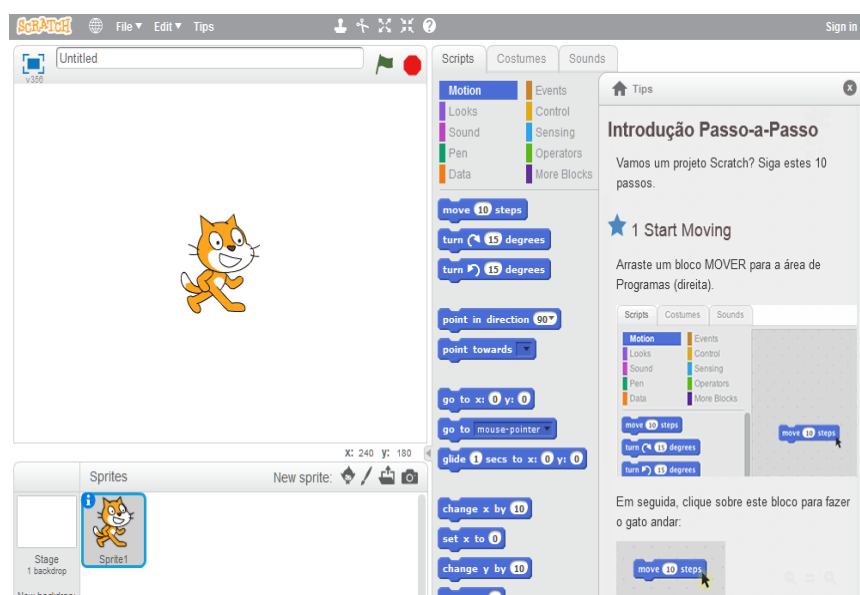


Figura 6.5. Tela do Programa Scratch 2.0 – Online

6.5. Scratch na prática: o desenvolvimento de habilidades e competências

No processo de criação com o Scratch, os alunos desenvolvem importantes habilidades e competências. De acordo com o “Programming Concepts and skills supported in Scratch”⁴, as competências para a resolução de problemas e a concepção de projetos se desenvolvem com o raciocínio lógico, a depuração de problemas, a produção de ideias, a atenção permanente e perseverante.

⁴ Programming concepts and skills supported in Scratch. Disponível em: <http://info.scratch.mit.edu/sites/infocrash.media.mit.edu/files/file/ScratchProgrammingConcepts-v14.pdf>. Acesso em: 09/08/2013.

Para Valente, o trabalho com linguagens de programação, neste caso o Scratch, pode colaborar para a criação autoral e a averiguação mediante as respostas do computador, desta forma “ele está sendo um elo importante no ciclo de ações descrição – execução – reflexão – depuração, que pode favorecer o processo de construção de conhecimento” (Valente, 2005). Assim, quando propiciamos aos alunos o desenvolvimento de atividades de forma desafiadora para ser executada através da produção de resultados advindos do uso de linguagem de programação, estamos solicitando muito mais que a elaboração de um produto, mas sim promovendo a oportunidade de execução de um ciclo contínuo, autoajustável e dinâmico de construção, reflexão e apropriação do conhecimento socialmente construído.

As etapas deste ciclo, que pode ser observada na Figura 6.6, ocorrem de forma natural e dinâmica durante todo o processo de utilização e criação com o Scratch, colocando o aluno num papel singular no processo de aprendizagem e de construção do conhecimento, mediado por computador. Este papel promove essencialmente a autoria e o protagonismo infantil e possibilita a manutenção de competências e habilidades imprescindíveis para a formação do cidadão que atua no século XXI, de forma crítica, ativa, reflexiva e construtiva.

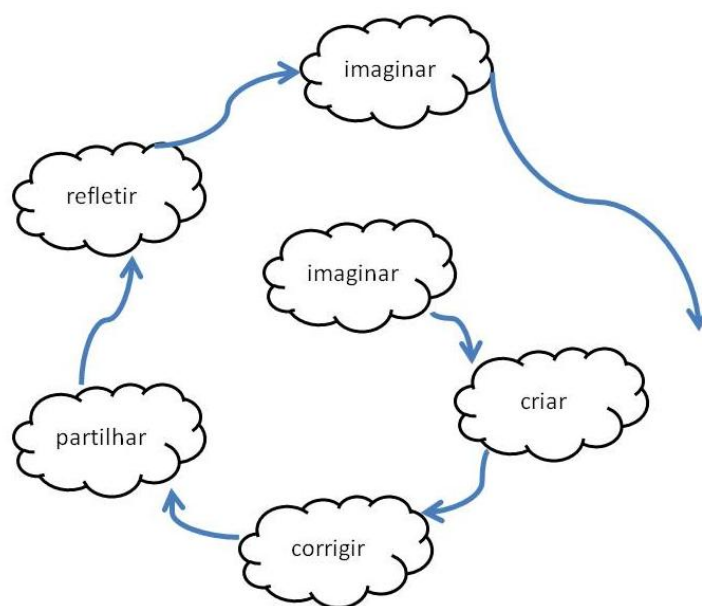


Figura 6.6. Organograma⁵: ciclo de aprendizagens

Para essa atuação, o aluno precisa colocar em jogo muito mais do que foi solicitado de forma padronizada na escola até então, pois, não se trata mais de afirmar e valorizar os conteúdos escolarizados e distantes da realidade social. É preciso oportunizar condições para que se possa colocar em jogo habilidades e competências amplas que abarcam o indivíduo na sua integridade e totalidade e que contemplem dentro das competências a cognição e o relacionamento tanto intrapessoal, como interpessoal, com objetivo de alcançar a capacidade de aplicação de seus conhecimentos em situações diversas, conforme pode ser visualizado no Tabela 6.1 a seguir:

⁵ Learning with Scratch – the Scratch Documentation Website-MIT. Disponível em: <http://info.scratch.mit.edu/sites/infoscratch.media.mit.edu/docs/Learning-with-Scratch.pdf>. Acesso em: 02/09/2013.

Tabela 6.1. Descrição das Habilidades e Competências para a formação do cidadão do século XXI

Habilidades	Competências
<ul style="list-style-type: none">- análise e sínteses;- planejar e avaliar suas ações;- interligar os conhecimentos adquiridos;- tomada de decisão	<p>Cognição: envolve estratégias e processos de aprendizado, criatividade, memória, pensamento crítico;</p> <p>Intrapessoal: a capacidade de lidar com emoções e moldar comportamentos para atingir objetivos;</p> <p>Interpessoal: envolve a habilidade de expressar ideias, interpretar e responder aos estímulos de outras pessoas.</p>
Transferência de conhecimento	
Capacidade de aplicar o que se aprendeu em situações novas	

Desta forma, trabalhar com o Scratch, de acordo com o “Learning with Scratch⁶”, favorece o desenvolvimento da fluência nas tecnologias digitais, onde, para ser fluente em uma língua exige-se a leitura, escrita e interpretação, sendo assim, para ser fluente em tecnologia digital é preciso aprender não apenas a interagir com o computador, mas também a criar com ele. A mesma postura que se espera do cidadão atuante e responsável, diante da nova configuração social, mediada pela tecnologia, que se forma no século XXI.

6.5.1. Possibilidades pedagógicas

São inúmeras as possibilidades pedagógicas com o uso do Scratch, o que contribui efetivamente para a promoção da contextualização curricular e da atribuição de significados aos conteúdos programáticos.

Dentre as mais conhecidas destacamos as animações, jogos de tabuleiro, jogos digitais, *games* literários, *quiz* e comunicação com outras interfaces (Android, Arduino e Makey Makey). Nas subseções a seguir, apontamos o diferencial e as contribuições de cada uma.

Animações - Criação de histórias animadas, com diálogos, efeitos sonoros e de movimento.

A animação é ideal para iniciar o trabalho de programação com Scratch nas séries iniciais do Ensino Fundamental, como o apresentado na Figura 6.7. Isto porque ela utiliza os recursos básicos do programa, é de fácil criação, possibilitando a observação rápida dos resultados. Facilmente encontra-se na *Internet* uma diversidade de tutoriais⁷ que auxiliam na criação de animações.

⁶ Learning with Scratch – the scratch documentation Website-MIT. Disponível em: <http://info.scratch.mit.edu/sites/infocratch.media.mit.edu/docs/Learning-with-Scratch.pdf>. Acesso em: 20/09/2013.

⁷ Untitled – Pensamento Digital. Disponível em: http://oficinas.pensamentodigital.org.br/ambientes_simulacao/scratch/primeiros_passos.pdf. Acesso em: 23/09/2013.



Figura 6.7. Exemplo de uma animação simples, criada nos primeiros contatos com Scratch⁸

Jogos de tabuleiro - Programação de jogos com temas e desafios gerados a partir da criatividade do programador.

A criação de jogos de tabuleiro utiliza outros recursos do Scratch que normalmente não são utilizados numa animação. Envolve o uso de sensores, operadores e, em alguns casos, variáveis. É possível verificar como utilizar estes comandos no “Tutorial Scratch – Conceitos básicos (versão XO-OLPC)”⁹, desenvolvido pela UNICAMP, que descreve a funcionalidade dos sensores, operadores e variáveis.

No exemplo a seguir, um jogo de tabuleiro criado e programado por alunos do Ensino Fundamental I de uma escola da rede pública municipal de São Bernardo do Campo - PMSBC (Figura 6.8). O jogo utiliza sensores e operadores, é possível verificar seus comandos acessando o compartilhamento no Scratch.mit.¹⁰

⁸ Dançarina by Elaine Rocha on Scratch – MIT. Disponível em: <http://scratch.mit.edu/projects/785207/>. Compartilhado em: 01/12/2009.

⁹ Tutorial Scratch – Conceitos básicos (versão XO-OLPC). Disponível em: <http://portalsme.prefeitura.sp.gov.br/Projetos/ie/Documentos/scratch.pdf>. Acesso em: 03/08/2013.

¹⁰ Percurso do Ambiente on Scratch – MIT. Disponível em: <http://scratch.mit.edu/projects/12774077/>. Compartilhado em: 30/09/2013.

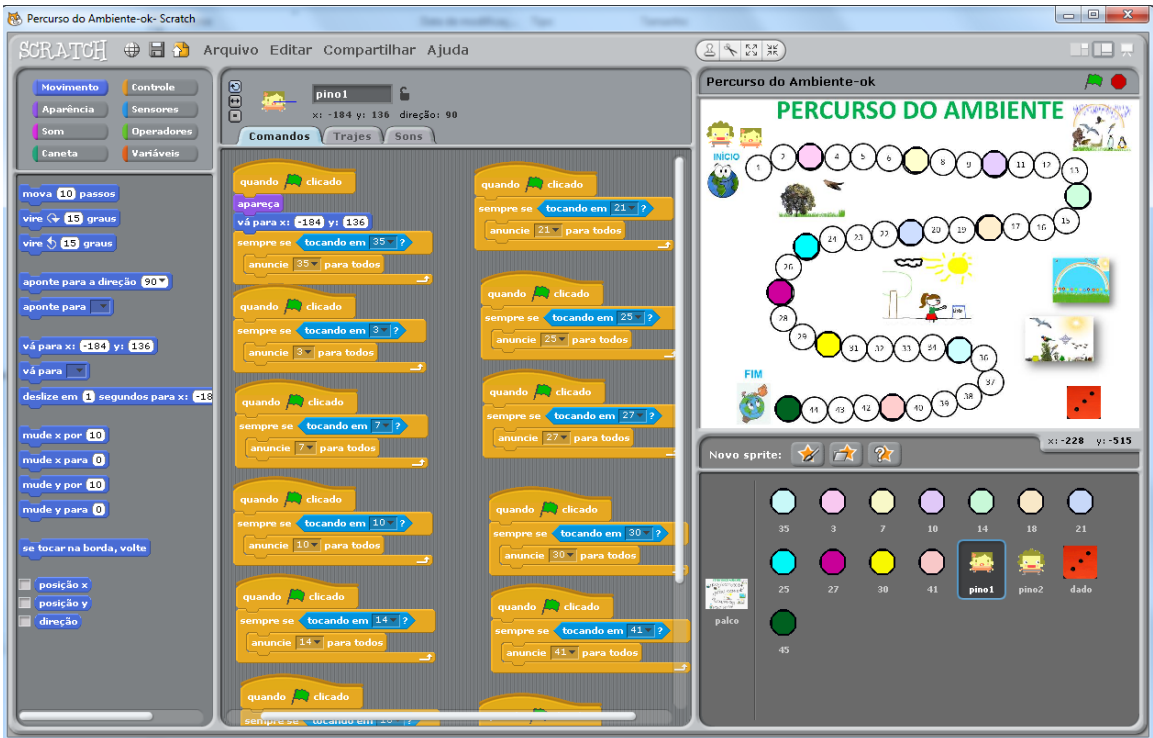


Figura 6.8. Jogo de tabuleiro criado por alunos do Ensino Fundamental I

Jogos ou games digitais - Criação de uma diversidade de jogos, com diferentes propostas, dentro das possibilidades de programação do Scratch.

Os jogos digitais são muito atraentes para o público infantil e adolescente. É a oportunidade de reproduzir seus *games* favoritos ou de criar um jogo de sua autoria, saindo do papel de consumidor para o de produtor de *games*. Assim que os alunos compreendem a utilidade dos diversos comandos, iniciam a produção, partindo dos mais simples, para os mais complexos, conforme vão entendendo a lógica dos diversos comandos.

A seguir, tem-se o exemplo de um *game* criado por um jovem participante da oficina “Criação de jogos com Scratch”, oferecida aos adolescentes da comunidade da EMEB José Cataldi (PMSBC), sendo este, o primeiro projeto desenvolvido pelo aluno (Figura 6.9).



Figura 6.9. Game criado pelo adolescente Henrique na oficina de Scratch para a comunidade¹¹

Games literários - Criação de *games* para divulgar uma obra literária, trabalhar com interpretação e criação a partir do estudo literário de uma obra existente e histórias de autoria dos alunos.

Estes *games* tornam-se um importante recurso para desenvolver um trabalho de programação aliado ao trabalho de literatura contemplado no currículo escolar. Ele pode abranger as diversas possibilidades de uso do Scratch, como no exemplo a seguir, em que apresentamos um *game* criado a partir do livro “A princesa escolhida” (Figura 6.10). Este *game* se inicia com uma animação, a qual é seguida de etapas que representam a sequência da narrativa desenvolvida na obra. Sua finalização instiga a leitura do livro para descobrir o desenrolar do conto.

¹¹ Caça às piranhas on Scratch. Disponível em: <http://scratch.mit.edu/projects/12774259/>. Compartilhado em: 30/09/2013.

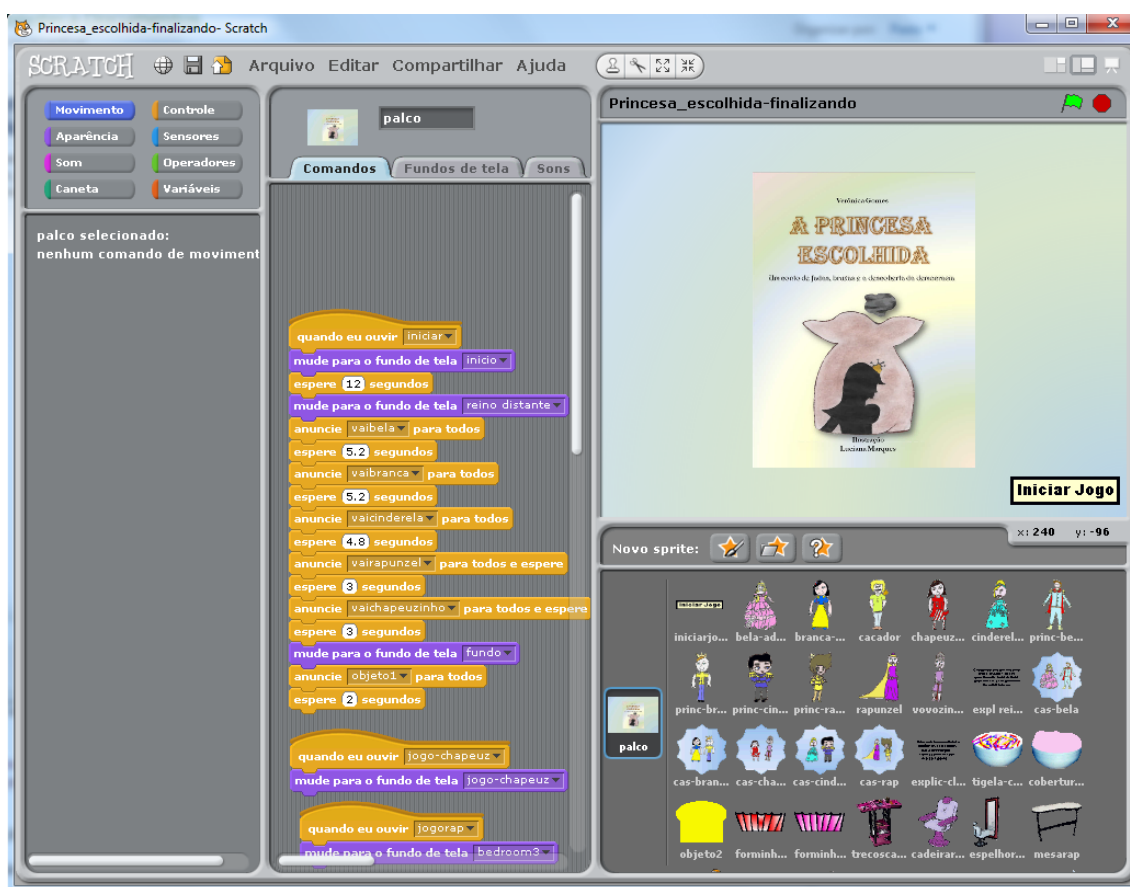


Figura 6.10. Game literário: "A Princesa Escolhida"¹²

Outra possibilidade de games literários são os criados a partir das histórias dos próprios alunos, como em "Isaac e Aleena" (Figura 6.11) criado pela dupla de alunos Victor e Gabriel, a partir da problemática existente no conto criado colaborativamente em sala de aula. Nesta proposta, cada dupla criou um game referente a um trecho do conto.

¹² Princesa_escolhida versão2 on Scratch. Disponível: <http://scratch.mit.edu/projects/13655707/>. Compartilhado em: 28/10/2013.

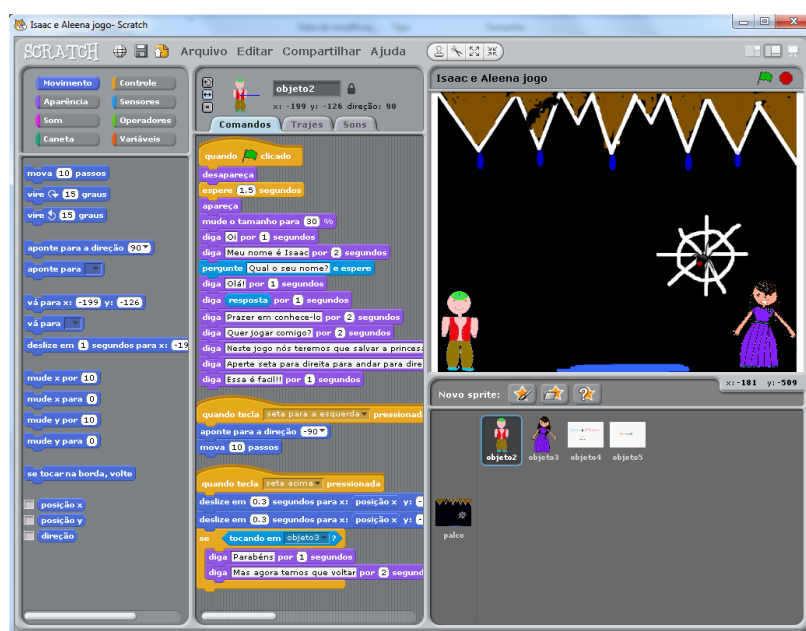


Figura 6.11. Game “Isaac e Aleena” criado a partir de um conto de autoria dos alunos

Quiz - Criação de *quiz* interativo

O *quiz* interativo pode ser uma oportunidade para o aluno criar desafios pensados a partir de um conhecimento construído. Para tanto, precisam elaborar perguntas e respostas possíveis para a interação das pessoas que serão os usuários do jogo. Pode ser criado com informações pesquisadas em um projeto ou, até mesmo, após o estudo de determinados temas. No exemplo a seguir, os alunos criaram um *quiz* para divulgar os parques da cidade (Figura 6.12), após um projeto que tinha, dentre outros temas, o estudo sobre as opções de lazer na cidade.¹³



Figura 6.12. Exemplo de *quiz* interativo elaborado por alunos

¹³ Conheça os Parques de SBC on Scratch. Disponível em: <http://scratch.mit.edu/projects/2178567/>. Compartilhado em: 27/11/2011.

No *site* oficial do Scratch.mit é possível encontrar diversos *quizzes*, como, por exemplo, o *Quiz* de Polígonos, criado para alunos da educação básica.¹⁴

Comunicação com outras interfaces:

Android - Possibilidade de integrar celulares Android na interação com jogos criados no Scratch

É possível controlar projetos no Scratch, utilizando dispositivos Android através da instalação do aplicativo Scratcher Control e o uso dos dispositivos (computador e celular) conectados através de uma mesma rede WiFi. Este aplicativo foi criado por Kreg Hanning e foi disponibilizado atualmente com opção em Português¹⁵. Ele permite a incorporação, de forma simples, de botões, gestos com os dedos, dados do acelerômetro, dados do sensor de luminosidade e comando de voz em projetos Scratch.

Por enquanto, este recurso funciona somente com a versão offline 1.4 do Scratch. A nova versão do Scratch 2.0, baseada em Flash, ainda não possui protocolo de comunicação com sensores externos.

Para configurar o aplicativo para uso, deve-se seguir os seguintes passos:

1. Instalar, no celular, o aplicativo do Scratcher Control;
2. No projeto Scratch, clicar com o botão direito sobre o bloco “valor do sensor” e habilitar conexões do sensor remoto, conforme demonstrado na Figura 6.13.

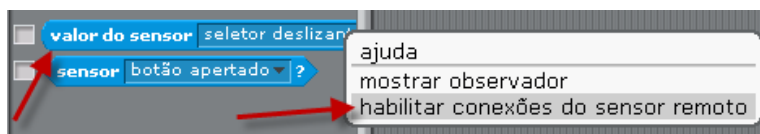


Figura 6.13. Habilitando valor do sensor

3. No Scratcher Control, inserir o endereço IP do computador que deverá estar na mesma rede (Figura 6.14).



Figura 6.14. Inserindo endereço IP

Para compreender o uso dos comandos de voz do Scratcher Control disponibilizados um tutorial online¹⁶ (Figura 6.15). Para utilizá-lo é necessário instalar o Scratcher Control no seu Android para executar os comandos.

¹⁴ Quiz polígonos on Scratch. Disponível em: <http://scratch.mit.edu/projects/3064603/>. Acesso em: 25/10/2013.

¹⁵ Scratcher Control- khanning.com. Disponível em: <http://khanning.com/scratchercontrol/>. Acesso em: 03/08/2013

¹⁶ Tutorial Scratcher Control no Scratch. Disponível em: <http://scratch.mit.edu/projects/12761186/>. Compartilhado em: 28/09/2013.

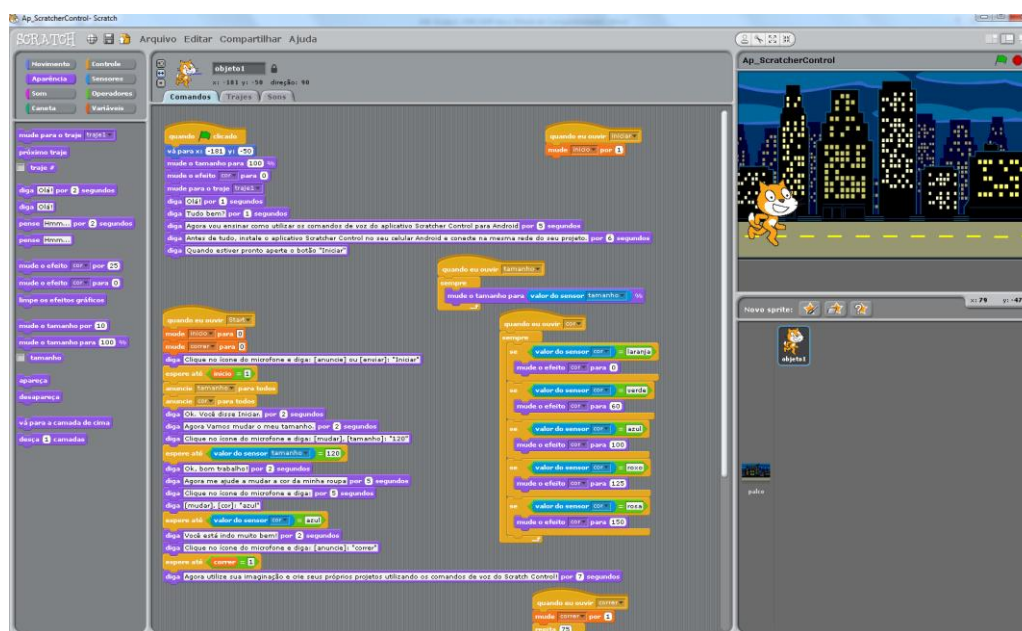


Figura 6.15. Instruções de utilização do comando de voz do Scratcher Control

Nas Tabelas 6.2, 6.3, 6.4 e 6.5 estão descritos os diversos comandos propiciados pelo Scratcher Control:

Tabela 6.1. Descrição dos comandos por gesto do Scratcher Control

Controle por Gestos	
Nome do Sensor	Significado em português-br
Comando Semelhante ao <i>TouchPad</i> do <i>Notebook</i>	
<i>Scroll_up</i>	Rolar para Cima
<i>Scroll_down</i>	Rolar para Baixo
<i>Scroll_left</i>	Rolar para Esquerda
<i>Scroll_right</i>	Rolar para Direita
Comando Semelhante ao Gesto Botão do <i>Mouse</i>	
<i>Tap</i>	Clicar
<i>Double_tap</i>	Duplo Clique
<i>Long_press</i>	Clicar e Segurar
Comando Semelhante ao Gesto ‘Cortar’	
<i>Flick_up</i>	Riscar para Cima
<i>Flick_down</i>	Riscar para Baixo
<i>Flick_left</i>	Riscar para Esquerda
<i>Flick_right</i>	Riscar para Direita

Tabela 6.2. Descrição dos comandos do joystick do Scratcher Control

<i>Joystick</i>	
Nome do Comando	Significado em português-br
<i>Button_a_pressed</i>	Botão A pressionado
<i>Button_b_pressed</i>	Botão B pressionado
<i>Button_x_pressed</i>	Botão X pressionado
<i>Button_y_pressed</i>	Botão Y pressionado
<i>Button_up_pressed</i>	Seta para Cima pressionada
<i>Button_down_pressed</i>	Seta para Baixo pressionada
<i>Button_left_pressed</i>	Seta para Esquerda pressionada
<i>Button_right_pressed</i>	Seta para Direita pressionada
<i>Select</i>	Botão Selecionar
<i>Start</i>	Botão Iniciar

Tabela 6.3. Descrição dos comandos do acelerômetro do Scratcher Control

<i>Acelerômetro</i>	
Nome do Comando	Significado em português-br
<i>Accelerometer-x</i>	Dados do eixo X do acelerômetro
<i>Accelerometer-y</i>	Dados do eixo Y do acelerômetro
<i>Accelerometer-z</i>	Dados do eixo Z do acelerômetro

Tabela 6.4. Descrição dos comandos do sensor de luminosidade do Scratcher Control

<i>Sensor de Luminosidade</i>	
Nome do Comando	Significado em português-br
<i>Light_level</i>	Dados do Sensor de Luminosidade

Arduino - Comunicação com Arduino através da modificação Scratch4Arduino, que possibilita a programação e funcionamento de objetos físicos

O Scratch4Arduino (S4A) foi desenvolvido, em 2010, pela equipe Smalltalk do Citilab.¹⁷ O S4A consiste em uma modificação do Scratch, que permite a programação simples do Arduino (plataforma de *hardware open source*)¹⁸, tornando possível a interação com seus diversos sensores e atuadores, podendo ser programado com as facilidades oferecidas pelo Scratch, sendo assim, acessível para alunos do Ensino Fundamental I.

De acordo com o *site* http://s4a.cat/index_es.html do Citilab, o S4A é compatível com o Arduino Diecimila, Duemilanove e Uno. O *Sprite* do Arduino encontra automaticamente a porta USB onde a placa está conectada, após a instalação do

¹⁷ S4A. Disponível em: <http://seaside.citilab.eu/scratch/arduino>. Acesso em: 03/09/2013.

¹⁸ Arduino Playground – Home Page. Disponível em: <http://playground.arduino.cc/Portugues/HomePage>. Acesso em: 03/09/2013.

firmware na placa Arduino. As instruções para instalação do *firmware* encontram-se disponíveis no *site* Citilab.¹⁹ Este *firmware* é um programa necessário para a placa Arduino se comunicar com o S4A. Para baixar e instalar o *software* do Arduino²⁰ basta seguir as instruções disponíveis no *site* oficial. Feita a instalação do *software*, o *download* do *firmware* e conectado o Arduino a uma porta USB do computador, basta abrir o arquivo *firmware* (S4AFirmware15.ino) no ambiente Arduino e, selecionar no menu ferramentas a versão correta da placa e a porta serial em que está conectada, enviando, em seguida, o *firmware* para a placa, utilizando o menu ferramentas – enviar. É possível ligar várias placas ao mesmo tempo, de acordo com a quantidade de portas USB disponíveis em seu computador.

Não é possível compartilhar projetos do S4A no *site* oficial Scratch.mit.edu, pois este não foi desenvolvido pelo MIT, não estando de acordo com as políticas de uso do *site*.

O Citilab configurou uma forma específica para a ligação das entradas e saídas dos componentes.²¹ Sendo assim, o S4A habilita seis entradas analógicas (pinos analógicos), duas entradas digitais (pinos digitais 2 e 3), três saídas analógicas (pinos digitais 5, 6 e 9), três saídas digitais (pinos 10, 11 e 13) e quatro saídas especiais para conectar servomotores de rotação contínua (pinos digitais 4, 7, 8 e 12). Também é possível a comunicação sem fio com uma placa por um módulo de RF.

Como exemplo, temos um projeto criado com o S4A, o qual utiliza um *fotoresistor* (LDR – *light diode resistor*) para interagir com o ambiente Scratch. Como não é possível compartilhar no ambiente Scratch.mit.edu, ele será descrito abaixo.

Para a execução do projeto foi feito as seguintes ligações na placa Arduino:

1. A extremidade ‘a’ do LDR foi conectada a saída 5v do Arduino e a extremidade ‘b’ a porta analógica 0.
2. Um resistor 10 ohms foi conectado entre a extremidade ‘b’ do LDR e a porta Gnd

Estas ligações podem ser melhor visualizadas na imagem abaixo (Figura 6.16):

¹⁹ Cambios S4A. Disponível em: http://s4a.cat/index_es.html e Arduino - Software. Disponível em: <http://s4a.cat/downloads/S4AFirmware15.ino>. Acesso em: 03/09/2013.

²⁰ Arduino – Software. Disponível em: <http://arduino.cc/en/Main/Software>. Acesso em: 03/09/2013.

²¹ Conforme informações obtidas em: http://s4a.cat/index_es.html

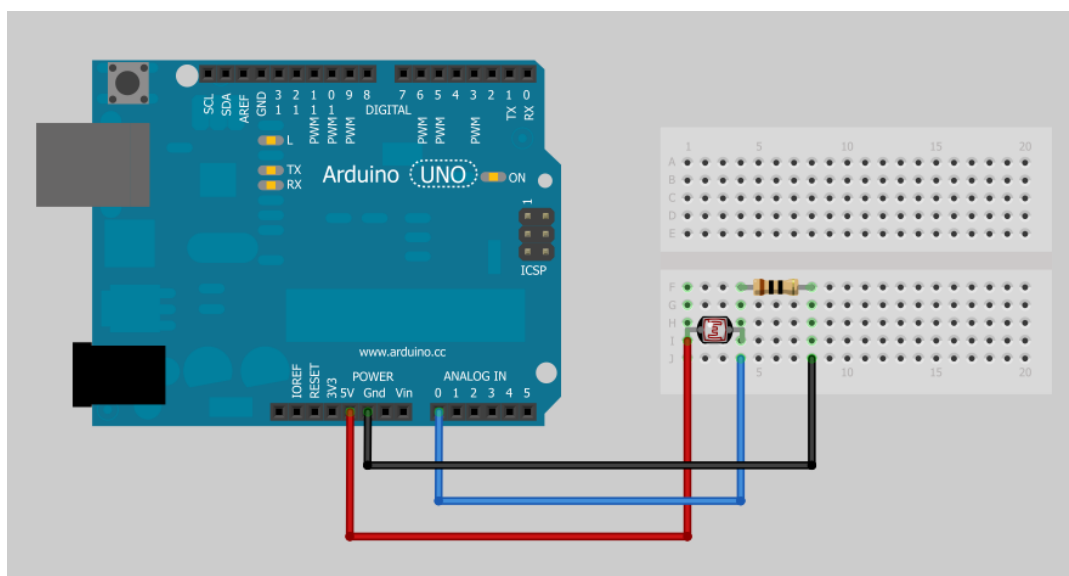


Figura 6.16. Ligações na placa Arduino

No Scratch foram realizadas as seguintes programações:

- Para o palco, demonstrado na (Figura 6.17):

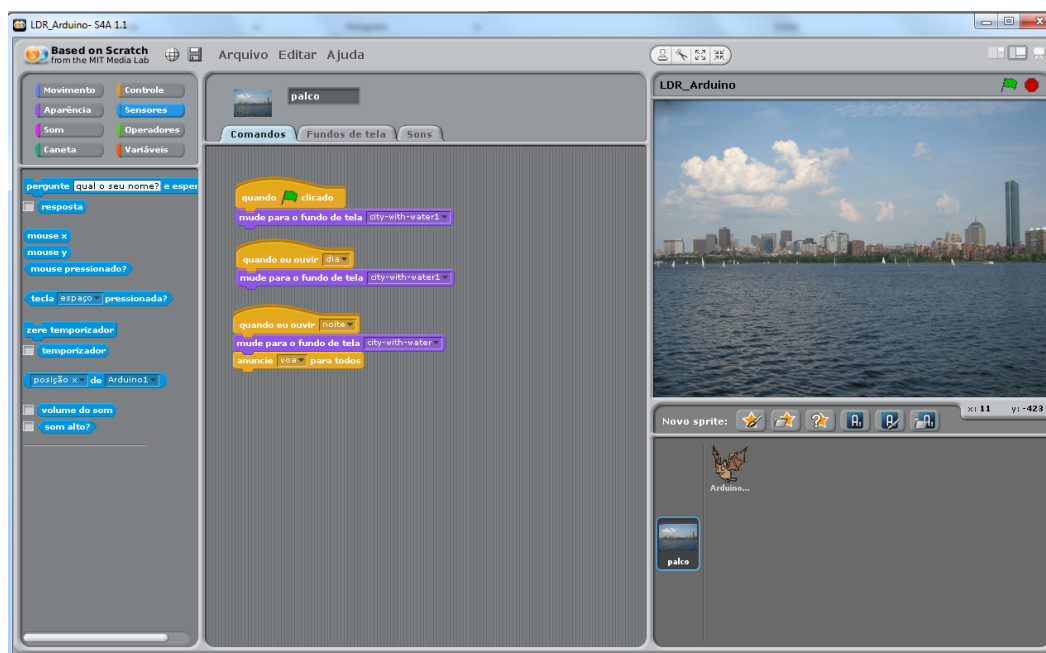


Figura 6.17. Tela com programação do palco no S4A

Foi configurada a mudança para o fundo de tela “dia” quando iniciado e, quando o sensor detectar o “anunciar dia” através de valores superiores a 50 (valor configurado de acordo com a iluminação local). A mudança para o fundo de tela noite foi configurada para acontecer quando o sensor “anunciar noite”, enviando também um comando para o morcego (objeto Arduino) voar, conforme é possível visualizar na figura a seguir os comandos utilizados (Figura 6.18):

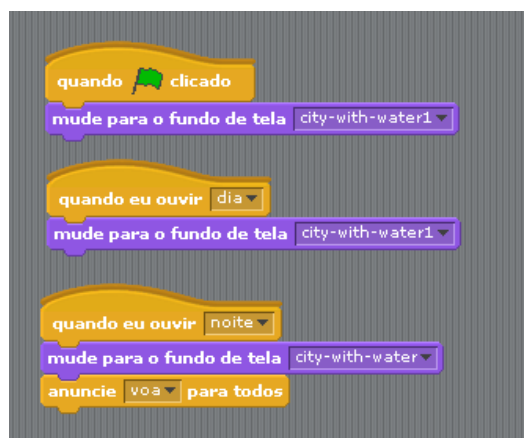


Figura 6.18. Programação definida para o palco no S4A

- Para o objeto (Arduino), conforme pode ser visualizado abaixo, foi escolhido dois trajes do morcego para o objeto existente e configurado os comandos (Figura 6.19):

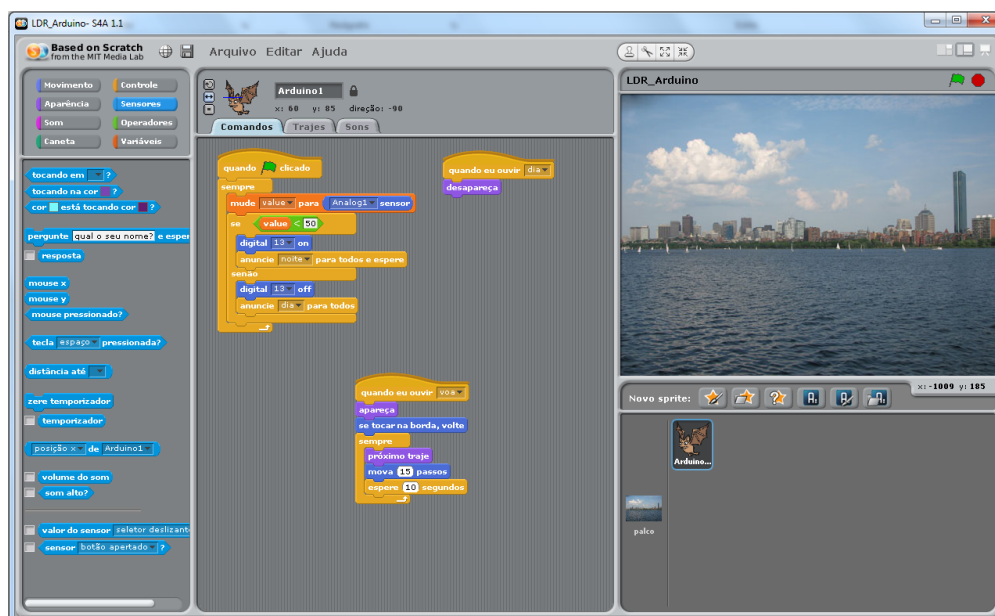


Figura 6.19. Tela do objeto "morcego" no S4A

O valor do sensor foi configurado para enviar o comando se receber valores menores que 50, anunciando noite, senão, enviará o comando anunciar dia, conforme é possível visualizar logo a seguir na Figura 6.20:



Figura 6.20. Programação definida para o objeto "morcego" no S4A

Como resultado teremos: Toda vez que o sensor detectar luz, mudará para o fundo de tela dia e o morcego desaparecerá, quando a iluminação diminuir, o Scratch mudará para o fundo de tela noite e o morcego irá se movimentar pela tela, conforme comandos mostrados nas imagens.

Makey Makey - Potencialização da interação propiciada pelo Makey Makey, o qual funciona como um teclado e oferece grande facilidade de funcionamento, além de uma infinidade de criações.

O Makey Makey foi criado por Jay Silver e Eric Rosenbaum, doutorandos no Media Labs do MIT. Consiste em uma pequena placa de circuito impresso com um microcontrolador Arduino ATMEGA32U4, que funciona substituindo o *mouse* e/ou o teclado, através de furos para conectar os fios, que por sua vez, poderão ser conectados a qualquer objeto ou substância que sua imaginação solicitar, basta que estes conduzam uma pequena quantidade de eletricidade, como por exemplo: massa de modelar, frutas, recipientes com líquidos, ou até mesmo pessoas. Tudo isto pode virar uma extensão do computador, permitindo interação.

Sobre o seu funcionamento, existem seis entradas na parte da frente da placa, que podem ser ligados, enquanto na parte de trás, há doze entradas sendo, seis para as teclas do teclado, e seis para o movimento do *mouse*, que pode ser acessado com *jumpers*. Se desejar usar um conjunto diferente de chaves, ou alterar as funcionalidades do Makey Makey, é possível reprogramá-lo usando o ambiente Arduino, também é possível utilizar vários Makey Makey juntos.

Quanto a sua utilização, ele dispensa qualquer conhecimento prévio sobre programação ou eletrônica, basta conectar os cabos e utilizar.

Como exemplo, temos um jogo que proporciona interação entre duas pessoas, criado em homenagem ao dia das avós²² (Figura 6.21)

²² Dia da vovó com Scratch e Makey Makey on Scratch. Disponível em: <http://scratch.mit.edu/projects/11617162/>.
Compartilhado em: 27/07/2013.

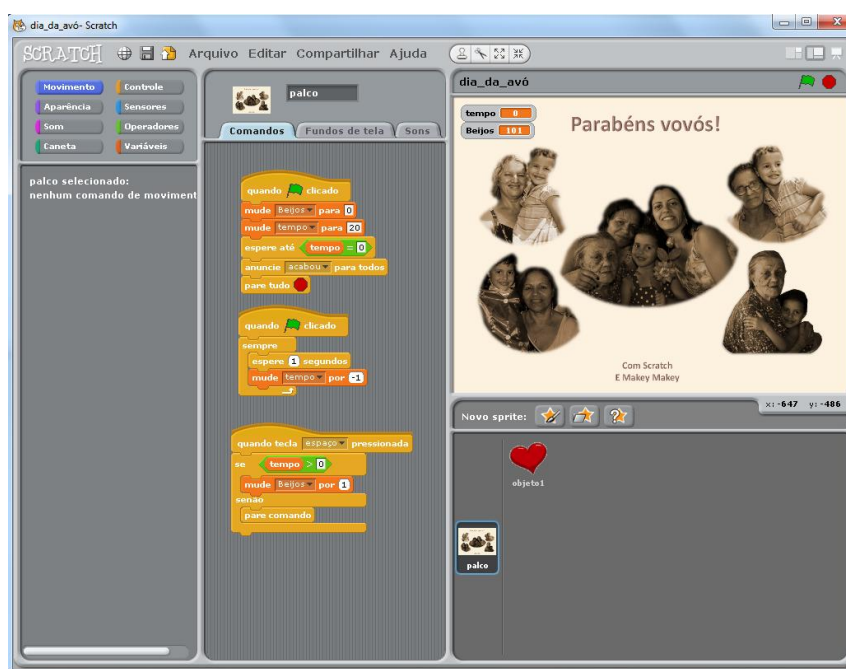


Figura 6.21. Interação propiciada pelo Makey Makey, associado ao Scratch

Caso não tenha o Makey Makey para testar o jogo, poderá visualizar seu funcionamento no vídeo a seguir: <http://www.youtube.com/watch?v=JPKD9BEC7FE>

Além de ser divertido e propiciar grande interação com diversos programas, o Makey Makey também pode ser utilizado como um recurso de acessibilidade, facilitando o manuseio e o acesso ao computador por pessoas com deficiência. Por exemplo, ao permitir que cada tecla de setas do teclado seja representada por um objeto e na distância ideal para cada um, uma pessoa que tem movimentos involuntários pode escolher o tamanho, o objeto e a distância, permitindo uma personalização para melhor atender suas necessidades.

6.5.2. Scratch e os recursos de acessibilidade

Os alunos com deficiência, ao utilizarem o Scratch, podem necessitar de algum recurso que viabilizem o seu uso. Os recursos utilizados para que a pessoa com deficiência tenha o acesso, neste caso ao Scratch, podem atender às necessidades visuais, motoras ou intelectuais. Alguns alunos com deficiência intelectual, por exemplo, necessitam de recursos visuais, com pouca informação, para melhor compreensão. Neste caso os comandos podem ser apresentados em cartões e retomados conforme a necessidade.

Outros podem ter a aprendizagem facilitada utilizando alguns *softwares*, como os apresentados na Tabela 6.6. O software Virtual Magnifying pode auxiliar os alunos com baixa visão e o Head Mouse pode ser utilizado por pessoas que não conseguem utilizar o *mouse* com as mãos.

Tabela 6.5. Software de acessibilidade

<i>Software</i>	<i>Característica</i>	<i>Uso</i>
Virtual Magnifying Glass ²³	Lupa virtual que amplia parte da tela	Facilita a visualização dos blocos de programação do Scratch por alunos com baixa visão
Head Mouse ²⁴	Comandos do <i>mouse</i> com a cabeça e expressões faciais	Indicado para pessoas que não conseguem utilizar o <i>mouse</i> convencional com as mãos e que tenham o controle da cabeça, olhos e boca.

Além destes *softwares*, há o recurso de acessibilidade do Windows em que é possível configurar o teclado com as funções do *mouse*, configurar o *mouse* para que o aluno não necessite manter o clique para arrastar um objeto, ampliar parte da tela, utilizar o teclado virtual, etc. O aluno também pode necessitar de protetores de teclas ou máscara de teclado (colmeia), *mouse* adaptado, *joystick* ou simples adaptações como regular a inclinação do teclado, colocar o monitor mais próximo do aluno, entre outros.

6.6. Considerações Finais

A discussão teórica, promovida à luz de referências autoral pertinente, trouxe o uso da linguagem de programação Scratch como um caminho que favorece a formação do cidadão do século XXI. Para tanto, foi resgatada a concepção de educação necessária para o uso contextualizado e significativo dos recursos tecnológicos na educação, tais como seus benefícios e diferenciais.

Como ponto central o Scratch foi foco de aprofundamento, sendo apresentado seu histórico de criação e possibilidades pedagógicas. Nesse item, destacamos a associação da linguagem de programação a outros periféricos como a plataforma Arduino e o Makey Makey, além de aplicativo para Android criado especialmente para o uso com Scratch. De forma mais simples, porém não menos importante, foram apresentados exemplos de atividades e de projetos criados com essa linguagem de programação contextualizados ao currículo, demonstrando seu potencial pedagógico e significativo. Foram apresentados, ainda, alguns recursos de acessibilidade que podem ser utilizados com alunos que deles necessitem.

Desta forma, entendemos que o Scratch – linguagem de programação escolhida para este trabalho – atende, de forma prática e conceitual o desenvolvimento de competências e de habilidades necessárias para a formação do cidadão atuante no século XXI. O contato com múltiplas linguagens (imagens, áudio, animações, jogos etc) favorece o desenvolvimento crítico e perspicaz na análise de mídias pelos alunos. O incentivo ao pensamento criativo e a curiosidade na busca de soluções inovadoras para problemas inesperados exigem uma atuação que requer uma postura autônoma, proativa, crítica, reflexiva, colaborativa, inclusiva e construtiva, uma vez que o mero consumo de conteúdos e programas já não é mais suficiente nesta sociedade. É preciso mais: transformar, remixar, criar, enfim, protagonizar no dia a dia fazendo a diferença.

²³ Virtual Magnifying glass 3.6 – source forge. Disponível em: <http://magnifier.sourceforge.net/#download>. Acesso em: 28/09/2013.

²⁴ Head Mouse – Indra – Tecnologias Acessíveis – Tecnologías Accesibles. Disponível em: <http://www.tecnologiasaccesibles.com/pt/headmouse.htm>. Acesso em: 28/09/2013.

Referências

- Ausubel, D. P. (1982) “A aprendizagem significativa: a teoria de David Ausubel”. Editora Moraes, São Paulo.
- Cavellucci, L. C. B. (2003) “Estilos de aprendizagem: em busca das diferenças individuais”. IA/Unicamp. Disponível em http://www.iar.unicamp.br/disciplinas/am540_2003/lia/estilos_de_aprendizagem.pdf Acesso em 20 de setembro de 2013.
- Lifelong Kindergarten Group, Massachusetts Institute of Technology Media Lab. “Programming concepts and skills supported in Scratch”. Disponível em: <http://info.scratch.mit.edu/sites/infoscratch.media.mit.edu/files/file/ScratchProgrammingConcepts-v14.pdf> . Acesso em 30 de julho de 2013.
- Lifelong Kindergarten Group, Massachusetts Institute of Technology Media Lab . “Learning with Scratch”. Disponível em: <http://info.scratch.mit.edu/sites/infoscratch.media.mit.edu/docs/Learning-with-Scratch.pdf>. Acesso em 30 de julho de 2013.
- Martins, A. R. Q. (2012) “Usando o Scratch para potencializar o pensamento criativo em crianças do Ensino Fundamental”. Dissertação de mestrado.Universidade de Passo Fundo. Disponível em <http://www.upf.br/ppgedu/images/stories/defesa-dissertacao-amilton-rodrigo-de-quadros-martins.PDF> Acesso em 01 de agosto de 2013
- Prensky, Marc. (2001) “Digital Natives, Digital Immigrants”. MCB University Press, Vol. 9 No. 5, October. Disponível em: <http://www.marcprensky.com/writing/Prensky%20-%20Digital%20Natives,%20Digital%20Immigrants%20-%20Part1.pdf>. Acesso em: 31 de julho de 2013.
- Resnick, M.; Rusk, N.Lifelong e Maloney, J. (2011) “21st Century Learning Skills” Kindergarten Group, Massachusetts Institute of Technology Media Lab.. Disponível em: <http://llk.media.mit.edu/projects/scratch/papers/Scratch-21stCenturySkills.pdf> . Acesso em 30 de julho de 2013.
- Silva, R. M. da. (2007) O uso da linguagem Logo na educação infantil. Artigo Acadêmico - Universidade Federal de Mato Grosso (UFMT). Disponível em <http://www.sucesumt.org.br/mtdigital/anais/files/OUsoDaLinguagemdeProgramacaoLogo.pdf>. Acesso em 30 de Julho de 2013.
- Souza, Clarisse S. de. (2013) Projeto leva ensino da programação às escolas. O Estado de São Paulo, Maio. Disponível em: <http://blogs.estadao.com.br/link/?s=clarisse%20de%20Souza>. Acesso em 24 de outubro de 2013.

- Surman, Mark. (2013) Alfabetização digital é porta de entrada para o século 21, Folha, Fevereiro. Disponível em <http://www1.folha.uol.com.br/tec/1223970-alfabetizacao-digital-e-porta-de-entrada-para-o-seculo-21-diz-diretor-da-mozilla.shtml>. Acesso em: 28 de julho de 2013.
- Valente, J. A. (2005) Pesquisa, comunicação e aprendizagem com o computador. O papel do computador no processo ensino-aprendizagem. Salto para o futuro. Disponível em <http://portal.mec.gov.br/seed/arquivos/pdf/1sf.pdf>. Acesso em 19 de Setembro de 2013.
- Vygotsky, L. S. (1991) A formação social da mente. Livraria Martins Fontes Editora Ltda. São Paulo. 4ª edição brasileira.