



Submission: dd/Mmm/yyyy; dd/Mmm/yyyy	1 st round notif.: dd/Mmm/yyyy;	New version: dd/Mmm/yyyy;	2 nd round notif.:
Camera ready: dd/Mmm/yyyy; dd/Mmm/yyyy	Edition review: dd/Mmm/yyyy;	Available online: dd/Mmm/yyyy;	Published:

Sistemas lúdicos de autoria no ensino-aprendizagem

Title: Authoring playful systems on teaching-learning

Luis Eduardo de Matos
Universidade Federal de Alfenas
luis.matos@bcc.unifal-mg.edu.br

Gabriel Gerber Hornink
Universidade Federal de Alfenas
gabriel.hornink@unifal-mg.edu.br

Augusto Márcio da Silva Júnior
Universidade Federal de Alfenas
augustomsj@gmail.com

Resumo

A crescente inserção das novas tecnologias digitais no cotidiano das pessoas tem modificado acentuadamente a forma de lidar com diversos problemas sociais, como por exemplo a utilização dessas ferramentas nos processos de ensino-aprendizagem. Os métodos tradicionais de ensino passam por um conflito ou uma concorrência gerada pela nova realidade multimídia oferecida aos alunos que, por sua vez, perdem gradativamente o interesse no aprendizado tradicional. Nesse contexto, a presente pesquisa tem como objetivo principal apresentar a construção e aplicação de uma unidade instrucional online, onde conceitos do pensamento computacional são postos em prática em atividades lúdicas a partir de ferramentas gratuitas, como as plataformas code.org e Scratch para programação em blocos (Blockly). Na pesquisa, a unidade instrucional online foi projetada no ambiente virtual de aprendizagem Moodle, sendo possível realizar discussões em fóruns e coletar informações referente a participação dos estudantes. Por fim, identificou-se que as ferramentas utilizadas proporcionam boas alternativas para a inserção de mídias digitais no ensino. Dessa forma, busca-se estreitar a relação educador/educando favorecendo a autoria dos mesmos na produção de seus próprios materiais e conhecimentos.

Palavras-Chave: Autoria; Code.org; Computação; EAD; ensino-aprendizagem; Lúdico; Scratch.

Abstract

The increasing insertion of new digital technologies in people's daily lives has significantly modified the way of dealing with various social problems, such as the use of these tools in the teaching-learning processes. The traditional methods of teaching go through a conflict or a competition with the new multimedia reality offered to students which, in turn, gradually lose interest in traditional learning. In this context, the present research has as main objective to present the construction and application of an online instructional unit, where concepts of computational thinking are put into practice in ludic activities available by free tools such as code.org and Scratch platforms for programming in (Blockly). In the research, the online instructional unit was designed in the virtual learning environment Moodle, which enables to hold discussions in forums and gather information regarding students participation. Finally, it was identified that the tools used provide good alternatives for the insertion of digital media in teaching. In this way, we seek to strengthen the teacher / student relationship by favoring their authorship in the production of their own materials and knowledge.

Keywords: Authorship; Code.org; Computing; EAD; teaching-learning; Playful; Scratch.



1 Introdução

Os avanços no desenvolvimento de tecnologias digitais possibilitou novas formas de lidar com as questões sociais, principalmente a educação, abrindo possibilidades para novos processos de interação humana. Recentemente, com o crescente uso desses dispositivos eletrônicos, os estudantes têm a possibilidade de ter autoria sobre seu próprio material, de modo ativo, interativo e na "palma de sua mão".

As temáticas envolvendo ensino-aprendizagem e a relação educador/educando, fazem parte da história da sociedade, ao ponto de levantar debates e discussões que englobam, em sua ideia central, como maximizar o aprendizado conjunto dos estudantes do ensino básico e diminuir a evasão causada por diversos motivos, dentre eles a falta de atratividades na escola.

Por muitos anos, o professor, como figura central da sala de aula, foi o detentor do conhecimento. Este era quem "transmitia" o conhecimento aos alunos de variadas formas, prioritariamente com métodos tradicionais que na maioria das vezes não mostravam-se como uma alternativa interessante para uma parcela significativa dos estudantes.

Para mudar essa rotina, os profissionais da educação tentam compreender os motivos que levam a esse bloqueio, buscando novos modos de utilizar o tempo na sala de aula de forma mais produtiva, criativa e inclusiva, levando em consideração as dificuldades de cada estudante e sabendo que o mundo conectado em rede gerou novos meios de interação entre as pessoas na sociedade.

Com o avanço das tecnologias digitais e a ampliação do acesso aos dispositivos móveis, uma quantidade significativa dos estudantes possuem acesso a internet e a novas tecnologias, o que muda de uma certa forma o relacionamento pessoal e conseqüentemente a forma de se aprender novos conteúdos.

Com isso, é possível perceber que o papel do professor na educação atual segue para caminhos diferentes. No contraste entre a digitalização e as falhas nos métodos tradicionais, o professor se vê na necessidade de criar práticas mais interativas e eficazes, visto que as tecnologias transformaram vários setores da sociedade e, com isso, os processos de ensino-aprendizagem terão de se adequar às mudanças.

Logo, sabendo-se que o acesso ao conhecimento está muito mais próximo da população nos dias de hoje, o objetivo do professor se torna, cada vez mais, o de ser um mediador da relação conhecimento/aprendizagem. É neste ponto que as novas tecnologias educacionais podem contribuir nas atividades de professores e alunos, colocando-os de frente a uma nova realidade que tende a potencializar os processos de ensino-aprendizagem, podendo motivar os alunos e ampliar a construção de conhecimentos de modo ativo, interativo e colaborativo.

Além disso, as novas tendências para o uso da informática no ensino apontam para o estudante como autor de seu próprio conhecimento, marcando o caminho para a inclusão digital e para a autonomia do estudante. Dessa forma, é necessário ter uma visão progressista diante dos métodos tradicionais que nos foram passados nos últimos anos e encarar uma nova realidade tecnológica digital.

Aliado a isso, a inclusão digital é um fator de inclusão social, quebrando as barreiras socioeconômicas impostas por nosso sistema. Assim, é possível utilizar-se dos espaços que a internet proporciona agora e, desta forma, inserir os jovens como protagonistas da sociedade do conhecimento (DIAS 2011).



Este artigo é resultado da experiência realizada em um projeto de pesquisa desenvolvido no final de 2016 e início de 2017, abordando o ensino-aprendizagem do pensamento computacional por meio da programação por blocos (blockly), de forma lúdica e criativa, usando sistemas como code.org e Scratch. Busca-se apresentar e discutir o modo de estruturação das atividades online no ambiente virtual Moodle, assim como obter indícios para compreender a aplicação das atividades, visando a formação do aluno como autor de seus próprios materiais.

2 Reflexões teóricas

2.1 Uso das TICs no ensino

As TICs (Tecnologias da informação e comunicação) são ferramentas essenciais para as principais atividades sociais da chamada nova era digital. O computador é uma das ferramentas em que uma das funções é transmitir e receber informações e, portanto, servir como um comunicador (Valente, 2013), ou seja, instrumento sociocultural que medeia as relações humanas.

De acordo com Valente (2013), “O computador pode ser usado também como ferramenta educacional. Segundo esta modalidade o computador não é mais o instrumento que ensina o aprendiz, mas a ferramenta com a qual o aluno desenvolve algo, e, portanto, o aprendizado ocorre pelo fato de estar executando uma tarefa por intermédio do computador”.

O cotidiano dos estudantes está repleto de novas e variadas tecnologias, tornando essa interatividade digital um obstáculo às sessões tradicionais de ensino que mostram-se insuficientes para motivá-los ao estudo. Contudo o ensino apoiado pelo uso de dispositivos digitais tende a sobrepor as abordagens obsoletas (Maschio, 2015).

Logo, o aprendizado adquirido por meio da construção de ideias apresenta um favorável potencial lúdico, já que o estudante se torna o autor do seu próprio material e conhecimento, promovendo o aprendizado, interesse, autonomia e despertando a criatividade do aluno (Freire 1996). Ressalta-se que há diversos recursos digitais que podem ser utilizados de forma gratuita.

Diversos autores (Almeida, 2001; Mercado, 2002; Moraes, 2000) reconhecem as potencialidades da área de Informática, concordando que os computadores podem favorecer sobremaneira ao processo de educação escolar, visto que agora as máquinas estão cada vez mais presentes no cotidiano.

Aliado a isso, o ensino de computação impulsiona a criação de empregos por meio da inovação dos processos na nossa economia. O conhecimento em tecnologias da informação é um requisito básico para que os estudantes possam garantir a sua inserção no mercado de trabalho, independente da área de estudo ou do setor profissional escolhido (Von Wangenheim, 2014).

Entretanto, para que as novas ferramentas sejam usadas de forma eficiente e o processo traga melhorias significativas, é necessário uma interação efetiva entre os envolvidos, tomando como base a figura do professor mediador. Segundo Ramos (2014), “os desafios da escola moderna face às aceleradas mudanças neste campo, deverá corresponder igual esforço das instituições de formação inicial assegurando que os novos professores estejam suficientemente preparados para as suas missões de ensino, incluindo uma formação científica e pedagógica sólida, moderna e atualizada”.



Em contrapartida, um novo modelo educacional com uso de tecnologias digitais gera uma motivação à aprendizagem, tendo em vista que os aparelhos eletrônicos já se tornaram bastante populares e acessíveis mesmo a pessoas de baixa renda.

De acordo com Moran (2000), “Com a internet podemos modificar mais facilmente a forma de ensinar e aprender tanto nos cursos presenciais como nos à distância”. As ferramentas digitais apresentadas neste trabalho indicam novas alternativas de se trabalhar na dificuldade de aprendizado dos alunos, oferecendo oportunidades de visualizar fenômenos científicos e conceitos abstratos.

2.2 Inclusão digital

A globalização dos processos tecnológicos fez com que a sociedade adquirisse novos conceitos aliados às mudanças que impactam a sua estrutura social/econômica. Nesse ponto, ressalta-se a importância da inclusão social por meio da utilização de dispositivos tecnológicos que fazem parte do cotidiano da nova sociedade digitalizada.

Inclusão digital é possibilitar ao indivíduo o uso das tecnologias para ampliar o desenvolvimento, potencializar sua autonomia, aumentar sua liberdade e aprimorar a democracia (Silveira, 2011). É, portanto, uma forma de incluir socialmente o indivíduo, tendo em vista a nova imagem de uma sociedade informacional que estabelece suas relações por meio do uso de tecnologias digitais e da internet.

Para Dias (2011), “Com o advento da internet e a popularização do computador, começaram a surgir na década de 1990, em países pobres, programas de inclusão digital para populações excluídas do acesso à comunicação mediada pelo computador”. Contudo, apesar da crescente expansão tecnológica e das quedas consideráveis no preço dos equipamentos, as desigualdades socioeconômicas constituem a grande barreira para a comunicação em rede (Silveira, 2008), além do modelo de banda larga no país, que acaba por não suprir a demanda de internet.

Segundo Silveira (2011), “No Brasil, a expressão “exclusão digital” passou a caracterizar o fenômeno das barreiras socioeconômicas, colocadas diante da maioria da população, para uso das tecnologias da informação, desde o final dos anos 1990”. Logo, sabendo desse bloqueio imposto pela questão econômica, busca-se ampliar os limites da inclusão social/digital de forma a incluir também aqueles que não podem financiar o acesso às ferramentas.

De acordo com Dias (2011), “Os laboratórios de informática são, para milhões de crianças e jovens, o primeiro contato com o computador e a internet”. Se de fato os meios de comunicação são essenciais para a inclusão do indivíduo na sociedade, o poder comunicacional é cada vez mais realizado pelo acesso e uso pleno das tecnologias digitais. Assim, tendo foco na cidadania e não somente na modernização, a rede torna-se um meio de participação ativa do indivíduo na sociedade. Silveira (2011) apud Castells (2009) sustenta que os meios de comunicação tornaram-se o espaço social onde o poder é decidido.

Por fim, o uso dos computadores em rede possibilitam a distribuição do poder da informação entre as pessoas, favorecendo a ampliação da autonomia dos indivíduos e da sociedade. Com isso, inclusão digital autônoma muda as exigências e o cenário político, uma vez que o espaço social dos debates é cada vez mais aberto e visível a todos indivíduos conectados (Silveira, 2011).



2.3 Sistemas de autoria

O estudante como autor de seus próprios materiais digitais, aliado a mediação do professor, é uma nova tendência educacional. Ao inserir o uso de tecnologias no ensino, cria-se possibilidades para que o cidadão em formação passe do tradicional e desgastado instrucionismo, para o construtivismo, adquirindo aprendizado mais lúdico e significativo.

O uso da informática na educação deve ser fundamentado em teorias e práticas que enfatizem o processo de construção do conhecimento pelo próprio aluno. A prática pedagógica pressupõe a atividade do estudante, sua capacidade de construção do conhecimento, bem como a ação direta do professor, no sentido de favorecer a ampliação do processo (Artifon, 1999).

O uso de dispositivos eletrônicos e sistemas de autoria na educação pode proporcionar, além do contato com novas tecnologias, a ampliação de estratégias para o desenvolvimento de habilidades cognitivas, expressão do pensamento, criação e imaginação, possibilitando a criança tornar-se criadora de conhecimento e não apenas receptora de informações (Pereira, 2005).

Os *softwares* de autoria devem ser pertinentes ao ambiente escolar, bem como a ideologia proposta, fácil de ser utilizado pelos envolvidos, servindo como um facilitador da construção do conhecimento por meio do desenvolvimento de projetos digitais lúdicos. Esses ambientes de aprendizagem devem fornecer conexões individuais e coletivas, permitindo o desenvolvimento de tarefas vinculadas à realidade do estudante, integrando diferentes áreas do conhecimento e favorecendo a multidisciplinaridade (Pinto, 2012).

Desta forma, Pinto (2012) ainda afirma que “o ambiente criativo deve propiciar ferramentas que viabilizem a autoria e a manifestação de idéias inovadoras permitindo o desenvolvimento de um espaço de aprendizagem que possibilite a manipulação de objetos e idéias e a interação dos estudantes entre si e com os professores”. Essa possibilidade propicia a manifestação da criatividade, tendo como fator principal a relação educador/educando como forma de estimular e desafiar o estudante na busca do seu desenvolvimento.

Em Pereira (2005), apresenta-se o uso de um ambiente de autoria chamado LEGAL. Este ambiente permite que a criança possa expressar sua criatividade e imaginação para criar e contar histórias com funcionalidades simples e intuitivas. A ferramenta foi aplicada em uma turma de crianças com idade entre quatro e sete anos, e se mostrou eficaz ao despertar o interesse e a criatividade dos envolvidos. Segundo o autor, “Esta ferramenta traz em si os benefícios de um ambiente de autoria, criando ambientes virtuais tridimensionais e a possibilidade de utilizar ambientes imersivos de múltiplas projeções”.

Portanto, o processo de aprendizagem é facilitado pela execução das tarefas propostas de forma lúdica, e pela busca por soluções eficientes. Artifon (1999) afirma que “o desenvolvimento é resultante de estruturas e reestruturas progressivas, mediante a ação do sujeito sobre a realidade e desta sobre o sujeito. O conhecimento resultará da relação sujeito/objeto, relação esta em que os dois se solidarizam em lugar de se oporem”.

3 Materiais e métodos



Esta pesquisa é de natureza aplicada, com objetivos exploratórios e descritivos, sendo caracteriza, principalmente, com procedimentos experimentais, com dados quanti-qualitativos. O objetivo principal foi o desenvolvimento, aplicação e avaliação de uma unidade instrucional *online* para o ensino de conceitos iniciais de programação.

Realizou-se um mapeamento prévio de trabalhos que utilizaram ferramentas lúdicas de autoria para auxílio ao ensino/aprendizagem e identificou-se que o Code.org e o Scratch apresentavam as melhores alternativas para a construção do curso, devido principalmente a facilidade no uso das mesmas, além da possibilidade de serem aplicadas no próprio navegador.

A unidade instrucional *online* foi ofertada no Moodle (ambiente virtual de aprendizagem) de forma livre e, portanto, aberta a toda comunidade e sem necessidade de conhecimento prévio. Os participantes eram das mais variadas idades, desde alunos do ensino fundamental até professores de escolas públicas e privadas. Contudo, a maior parte dos concluintes foram educadores que buscavam adquirir conhecimento em novas tecnologias que pudessem ser utilizadas na sala de aula, auxiliando os processos de ensino.

A unidade instrucional foi ofertada em duas etapas distintas. Na primeira etapa foi utilizado as plataformas Code.org e Scratch, sendo aplicada entre Novembro de 2016 e Janeiro de 2017, contando com 80 inscritos. Na segunda etapa utilizou-se apenas da plataforma Scratch, sendo aplicada entre Abril e Maio de 2017, contando com 319 inscritos.

3.1 Levantamentos prévios

Os mapeamentos prévios tiveram como objetivo conhecer o que existe no Brasil sobre o uso de ferramentas digitais de autoria e, assim, identificar as possibilidades que cada uma poderia agregar a construção e aplicação da unidade instrucional online. Segue no apêndice E uma tabela com alguns softwares de autoria encontrados por buscas feitas nos mecanismos de pesquisa do Google.

Outro levantamento levou em consideração os cursos de Scratch disponíveis no momento que o projeto estava em andamento, coletando-se dados como público alvo, duração, ferramentas utilizadas, etc. Segue na tabela 1 alguns cursos em vigência no período do projeto.

Tabela 1: Cursos de Scratch abertos no período da realização do projeto

Instituição	Ferramenta	Público	Gratuito?	Tipo	Duração
PUC-SP	Scratch/ Arduino	Educadores	Não (342,00)	Presencial/ interativo	8 horas
IFES	Scratch	Graduação/ Técnico	Sim	EAD/ Moodle	45 horas
Unifenas	Scratch	Geral	Sim	EAD/ autoguiado	4 aulas
UNDB	Scratch	fundamental	Não (350,00)	Presencial/ interativo	10 horas/ 5 aulas

Fonte: Os autores

3.2 Scratch e Code.org



As mudanças sociais causadas pela expansão tecnológica do século atual apontam como uma nova tendência mundial a inclusão do ensino de computação nas fases escolares iniciais (Da Cruz Alves, 2016). Diversas iniciativas surgem agora nesse sentido, como, por exemplo, o uso do Code.org e do Scratch no ensino de crianças e adolescentes.

O Code.org é uma organização sem fins lucrativos criada em 2013 cujo objetivo é divulgar e disseminar o ensino de programação para todo tipo de pessoa e idade, dando oportunidade para que escolas tenham um material interativo disponível sobre a área. A instituição acredita que o ensino de computação deve fazer parte do currículo escolar e possui parcerias com vários gigantes da tecnologia, tendo algumas das aulas ministradas por personalidades da área como Mark Zuckerberg e Bill Gates.

Com uma plataforma de desenvolvimento *online*, disponível na página web <https://br.code.org>, a hora do código no Brasil pretende mudar a ideia tradicional de que programação seja algo muito avançado e difícil, permitindo que pais, professores e alunos aprendam programação de forma intuitiva e divertida.

Durante o primeiro módulo da primeira oferta do curso, foram utilizadas algumas atividade da "Hora do Código" para introduzir os envolvidos aos conceitos básicos de programação, assim ao modo de se construir o código no code-org, que é similar ao Scratch.

O Scratch foi desenvolvido pelo *Lifelong Kindergarten Group* no *MIT Media Lab* em 2007 e hoje se apresenta em vários idiomas, alguns distribuídos como *software* gratuito (<http://scratch.mit.edu>), outros como *software* livre (<http://snap.berkeley.edu>). É um sistema que pode ser utilizado para os primeiros contatos com conceitos iniciais de raciocínio lógico por intermédio da programação.

Com o Scratch é possível criar personagens que dançam, cantem e interagem uns com os outros, apenas arrastando e soltando blocos. Esses blocos representam os componentes do programa, como expressões booleanas, loops, condicionais e variáveis. A plataforma possui também uma rede social com projetos para compartilhamento.

O Scratch possui uma comunidade de aprendizagem criativa presente em mais de 150 países com mais de 11 milhões de usuários e quase 4 milhões de projetos compartilhados. Diversas pesquisas mostram que o uso do Scratch contribui positivamente no ensino de computação nas escolas (Von Wangenheim, 2014)

Este sistema de autoria foi utilizado no segundo tópico da primeira oferta do curso *online* e também durante toda a segunda oferta, quando foram aprofundados os conceitos de programação e o estudante pode construir seus próprios objetos, de acordo com as áreas de interesse do mesmo.

As ferramentas de desenvolvimento de aplicativos, como por exemplo o App Inventor (criado pela Google em 2010 e vendido ao MIT em 2013), são de modo geral de fácil entendimento para aqueles que não estão familiarizados com os conceitos de programação, uma vez que são construídos para uso intuitivo, baseado em cores, encaixes e semânticas. Para Finizola (2014), “O ensino de programação é importante, e não deveria ser uma atividade restrita apenas a estudantes de computação”.

Diversos trabalhos apontam para a importância da inserção do pensamento computacional e da lógica de programação para o desenvolvimento de alunos em idade escolar, como dito em (Scaico et al, 2013, p.93) “este tipo de educação permite o desenvolvimento de diversas capacidades que contribuem para melhorar o raciocínio lógico dos estudantes”.

Além disso, o uso de ferramentas de desenvolvimento de aplicativos no ensino pode contribuir para apresentar ao estudante a oportunidade de usar a sua tecnologia disponível para



outras atividades além daquelas que estava acostumado a realizar. Aliado a isso, apoia-se o desenvolvimento de habilidades de raciocínio lógico e resolução de problemas (Finizola, 2014).

Finizola (2014) apresenta em seu trabalho um curso de ensino de programação para 20 alunos do 1º ano do ensino médio de uma escola pública. A ferramenta escolhida foi a plataforma App Inventor, alegando-se que a mesma apresenta conceitos de programação de modo atraente e significativo para os alunos, além de abstrair a complexidade da programação tradicional.

Ainda segundo Finizola (2014), “A turma mostrou-se motivada e ansiosa em ver o aplicativo desenvolvido executando em seus próprios tablets”. A plataforma App Inventor foi apropriada para atingir o objetivo proposto pelo curso, possibilitando um feedback rápido da construção dos algoritmos.

Uma outra unidade instrucional foi aplicada em uma turma de 24 alunos do 1º ano do ensino fundamental de escola privada, utilizando a plataforma Scratch. De acordo com Von Wangenheim (2014), “as crianças desenvolveram uma versão interativa da história infantil Chapeuzinho Vermelho, de forma integrada ao conteúdo das disciplinas de Língua Portuguesa e Artes”.

Concluiu-se com os resultados da pesquisa que o uso do Scratch possibilitou a aprendizagem dos conceitos básicos de computação de forma efetiva, divertida e lúdica, despertando o interesse e motivação dos alunos pela tecnologia digital (Von Wangenheim, 2014). Dessa forma, ressalta-se o quanto esse tipo de atividade contribui para motivação escolar, aprendizagem de conceitos escolares específicos, do estímulo ao raciocínio lógico, e outras habilidades cognitivas.

3.3 Ambiente virtual de aprendizagem - Moodle

Os ambientes virtuais de aprendizagem tem chamado atenção devido a grande importância que exercem na área da educação, principalmente quanto ao seu suporte à aprendizagem colaborativa. Esses ambientes focam no trabalho em grupo, com vistas a promover a aprendizagem coletiva, através da colaboração de cada indivíduo participante (Bittencourt, 2007).

O aspecto social desses ambientes possibilita a conectividade como forma de compartilhar ideias, criar conteúdos dinâmicos e proporcionar o aprendizado coletivo (Serrão, 2016). Ainda, segundo Serrão (2016), “o aprendizado pode ser visto como um processo de formação de redes que é o ato de representar, conectar e organizar nodos especializados para facilitar a informação e o fluxo de conhecimento”.

O Moodle foi desenvolvido para possibilitar a criação de cursos a distância publicados na web em 2001 (versão 1.0), possibilitando interação com os envolvidos de forma mais prática e fácil. É distribuído como *open source*, licenciado sob a GNU GPL (General Public License - Licença Pública Geral GNU).

O sistema é utilizado por meio de um navegador web ou aplicativo e consiste em um conjunto de unidades funcionais denominadas módulos. Cada módulo então é responsável pela execução de determinadas funções, tais como criar cursos *online* e fóruns para discussões, além de permitir a criação e execução de testes.

O ambiente Moodle é uma plataforma essencial para o desenvolvimento deste projeto pois, além de possibilitar a criação dos módulos do curso de forma interativa e eficaz, dispõe de diversas ferramentas que foram utilizadas para coletar dados importantes dos participantes como



o número de atividades realizadas, as temáticas que tiveram maior foco e a frequência de participação, dentre outros.

3.4 Organização da unidade instrucional online

A unidade instrucional online foi desenvolvida e organizada de acordo com os principais temas e conceitos da programação computacional como operadores, variáveis, condicionais e repetição. Na primeira oferta do curso foi feita uma introdução aos conceitos computacionais utilizando atividades da plataforma Code.org (Figura 1), já que esta apresenta tarefas simples de trabalhar e semelhantes a programação que se utiliza no Scratch.

Em cada módulo do curso, uma sequência didática foi projetada para ser seguida de acordo com o tema a ser trabalhado na presente semana. Desta forma, cada módulo relacionado com seu respectivo tema possuía instruções e atividades semelhantes aos módulos anteriores, fazendo com que o estudante criasse uma maior familiaridade com a estrutura do curso e assim tivesse maior facilidade para entender o que deveria ser feito em cada semana. Um exemplo dessa sequência é apresentado na Tabela 2.

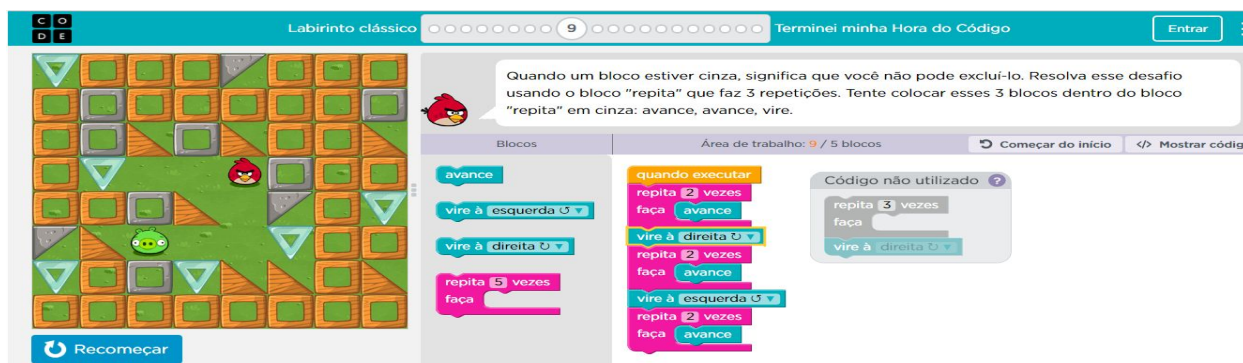


Figura 1: Exemplo de atividade introdutória utilizando a plataforma Code.org

Tabela 2: Exemplo de sequência didática a ser seguida em determinado módulo.

TEMA: Repetição SUBTEMA: Variantes das estruturas de repetição
ATIVIDADES: <ul style="list-style-type: none">● Realizar a leitura do material auxiliar introdutório ao conceito.● Assistir as vídeo-aulas de cada subtema.● Responder ao Quiz sobre a dificuldade encontrada em cada subtema.● Responder aos questionários de avaliação sobre o conteúdo dos subtemas.● Participar obrigatoriamente do fórum de discussão sobre o tema.

Fonte: Os autores

Com o intuito de exemplificar de forma mais clara, apresenta-se na Figura 2 um recorte de uma das telas da unidade instrucional online. Seguindo a sequência didática estabelecida, após cada vídeo aula (Figura 3) o participante realizava uma atividade avaliativa sobre o



conteúdo estudado anteriormente, com o intuito de fixar o tema visto. Esse tipo de atividade foi configurado no ambiente Moodle para que ficasse disponível apenas quando a restrição fosse cumprida, como nesse caso a visualização prévia da video aula.

Após uma breve introdução aos conceitos computacionais utilizando Code.org, na sequência da primeira oferta do curso utilizou-se o Scratch dando mais liberdade a criatividade do estudante nas criações que as atividades exigiam. No final, um trabalho prático de encerramento foi proposto (Figura 4), e seu objetivo era o desenvolvimento de um jogo utilizando todos os conceitos estudados no decorrer do curso.

Na segunda oferta da unidade instrucional *online*, as características da primeira oferta foram mantidas, retirando apenas a parte introdutória do Code.org e aperfeiçoando algumas atividades e tarefas. Dessa forma, dando um foco maior no Scratch, o curso apresentou um cenário mais atrativo e objetivo, o que facilitou uma maior participação dos estudantes em tarefas mais desafiadoras e abertas a criatividade dos mesmos.

The screenshot shows a Moodle course interface. At the top, there is a navigation bar with tabs for 'Boas vindas', '1ª Semana', '2ª Semana' (selected), '3ª Semana', '4ª Semana', '5ª Semana', and 'Projeto Final'. Below the navigation bar, the page title is '2ª Semana' and the subtitle is '11 até 25 de Abril - Repetição'. A large circular icon with a refresh symbol is on the left. To its right, the text reads: 'Neste módulo trabalharemos a estrutura de repetição simples e composta no scratch, pratiquem bastante e utilizem a criatividade. Vamos juntos?'. Below this, there is a list of activities with checkboxes on the right: 'Leia e siga as orientações.' (checked), 'AULA 04 - Repetição no Scratch' (checked), '4.1 - Leitura - Estrutura de Repetição no Scratch' (checked), '4.2 QUIZ- Após utilizar a estrutura de repetição no scratch, responda' (unchecked), '4.3 Avaliação - Questionário sobre o vídeo 04' (unchecked), and 'AULA 05 - Criando animações com novos atores' (checked). A 'Restringido' (Restricted) label is present, with the text: 'Disponível se: A atividade AULA 04 - Repetição no Scratch esteja marcada como concluída'.

Figura 2: Recorte de uma da tela da 2ª semana, que tratou do conceito de Repetição

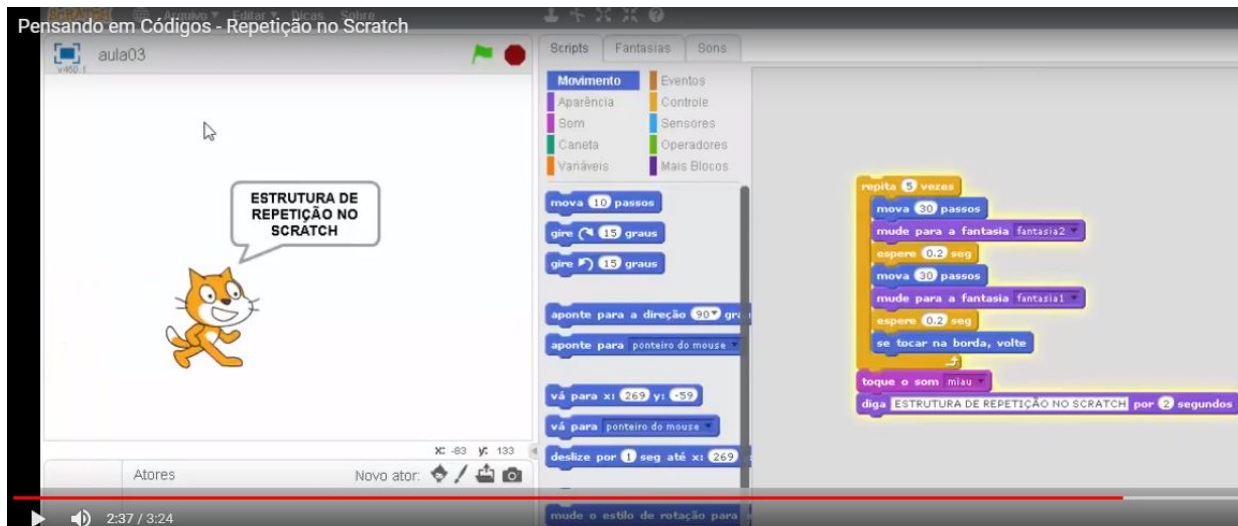


Figura 3: Recorte de uma vídeo aula sobre o conceito de Repetição

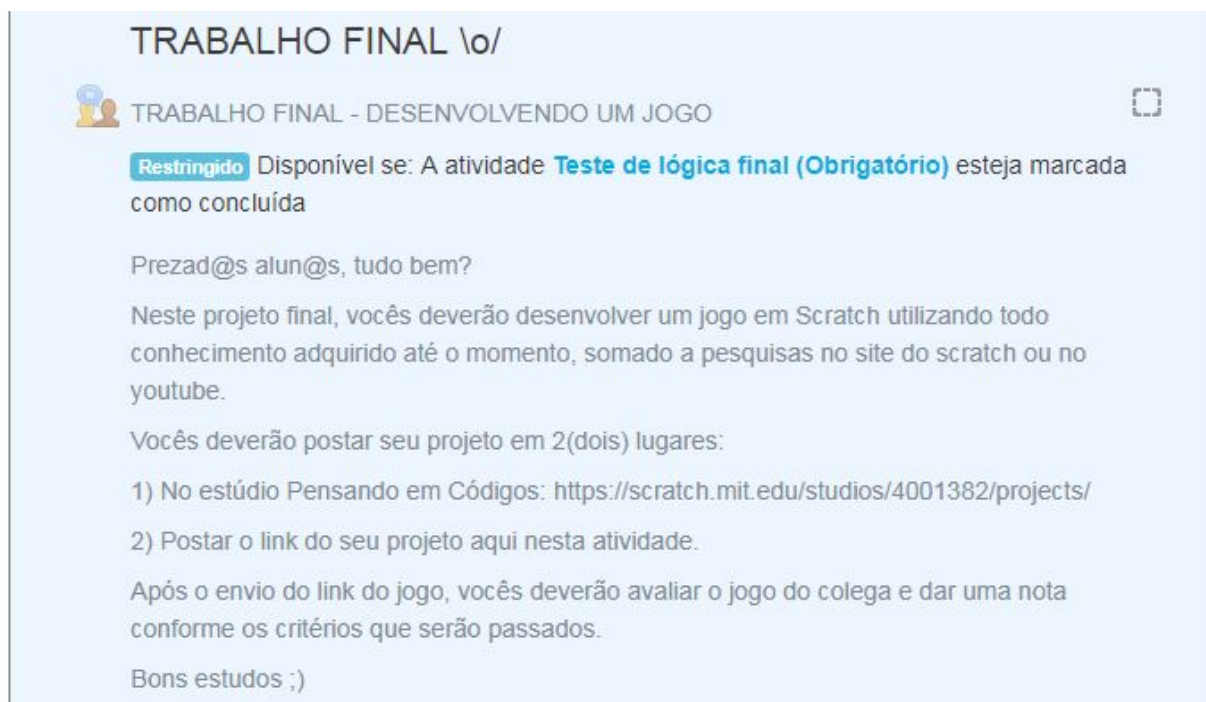


Figura 4: Recorte da tela do projeto final, desenvolvimento de um jogo

3.5 Obtenção, sistematização e análise dos dados

A análise foi feita por triangulação dos dados, incluindo análise estatística descritiva dos dados quantitativos e correlação dos dados dos questionários com os resultados dos projetos. A análise estatística buscou identificar alguma melhoria nos dados pós curso em relação ao conhecimento de programação e Scratch.



Os dados foram obtidos prioritariamente utilizando as funcionalidades disponíveis no ambiente Moodle, incluindo:

1. Registro de Log do Moodle (N° de atividades concluídas e visualizações)
2. Registro dos Fóruns (Participação em dúvidas, respostas e discussões)

Além disso, foram elaborados questionários de avaliação pré e pós curso para identificar os conhecimentos prévios e posteriores ao curso em relação programação e aos aplicativos utilizados. Vale ressaltar que foram coletados os resultados apenas daqueles que concluíram no mínimo 75% das atividades do curso. Incluiu-se:

1. Questionário pré-curso: Conhecer o nível de conhecimento em programação e em Scratch, através de avaliação dos próprios participantes, numa escala de 1 a 5.
2. Questionário pós-curso: Avaliação da melhora do conhecimento em programação e em Scratch, segundo os próprios participantes, numa escala de 1 a 5.

4 Resultados e Discussões

Por meio das funcionalidades que a ferramenta Moodle oferece, foi possível coletar diversos dados que nos possibilitaram avaliar e discutir a participação e a evolução dos estudantes que se envolveram com o curso. Por meio desses dados, busca-se entender os pontos positivos e negativos presentes na construção do curso, na interação com os estudantes, e no formato utilizado para a realização das atividades.

4.1 Medida de participação geral

A primeira oferta da unidade instrucional “Pensando em códigos” foi elaborada com um total de 87 atividades baseadas no Scratch e Code.org, divididas entre fóruns, quizzes, vídeo-aulas, materiais didáticos, e questionários avaliativos. Os participantes foram divididos em grupos com o intuito de facilitar a atuação dos formadores e propiciar melhores diálogos nos fóruns.

Analisando os registros de log do moodle, identificaram-se, para cada participante, quais atividades foram concluídas, possibilitando-se agrupar os mesmos em quatro faixas de percentual de realização de atividades (Menos de 25% das atividades concluídas até Mais de 75% das atividades concluídas), como apresentado na Tabela 3.

Tabela 3: Número de concluintes por etapa na 1° oferta da unidade instrucional.

Grupo	Menos de 25%	Entre 25 e 50%	Entre 50 e 75%	Mais de 75%	Total
Alpha	11	3	0	0	14
Beta	23	2	1	8	34
Gama	21	3	3	5	32
Total	55	8	4	13	80

Fonte: os autores



A partir desses dados, percebe-se que de um total de 80 participantes, apenas 13 (16,25%) chegaram a concluir mais de 75% do curso, valor considerado ideal para abranger todos os temas propostos. Esta quantidade de concluintes está dentro do esperado, tendo em vista que a taxa de abandono em MOOCs (*Massive Open Online Course*) é de 75% a 95% dos inscritos (Poy e Gonzales-Aguilar, 2014).

De forma semelhante, porém utilizando apenas a ferramenta Scratch, a segunda oferta da unidade instrucional “Pensando em Códigos” foi construída com 70 atividades. No total, foram 319 inscritos, divididos em 11 grupos. Na tabela 4 apresenta-se em detalhes a distribuição de conclusões (Menos de 25% das atividades concluídas até mais de 75% das atividades concluídas).

Tabela 4: Número de concluintes por etapa na 2º fase do curso.

Grupo	Menos de 25%	Entre 25 e 50%	Entre 50 e 75%	Mais de 75%	Total
Alpha	18	6	1	4	29
Beta	16	3	3	7	29
Capa	19	4	2	4	29
Delta	14	4	5	6	29
Epsilon	18	4	1	6	29
Eta	20	3	1	5	29
Gama	18	1	5	5	29
Iota	17	4	3	5	29
Lambda	22	0	4	3	29
Teta	20	3	2	4	29
Zeta	18	3	2	6	29
TOTAL	200	35	29	55	319

Fonte: os autores

Diante deste cenário, percebe-se que dos 319 participantes inscritos, 55 (17,25%) concluíram no mínimo 75% das atividades propostas, quantidade considerada ideal para o acompanhamento de todos os temas presentes no curso. Assim como mostrado na 1º oferta, o número de concluintes também está dentro do esperado, tendo em vista a taxa média de conclusão de MOOCs.

4.2 Medida de participação nos fóruns de dúvidas

Por meio do número de iterações obtido nos fóruns de dúvidas de cada módulo, é possível identificar aqueles que tiveram maior participação. Dessa forma, pode-se dizer que os temas mais discutidos foram aqueles que causaram maior dúvida ou despertaram maior interesse dos participantes.

Observa-se na tabela 5 que os fóruns que apresentaram maior número de interações na primeira oferta da unidade instrucional foram o de “Dúvidas - Módulo 1”, que tratou das



dúvidas iniciais, como cadastro na plataforma Code.org e atividades prévias, e o de “Dúvidas - Módulo 5”, que trabalhou com discussões sobre o conceito de repetição no Scratch.

Na segunda oferta, estas informações podem ser consideradas ainda mais consistentes, visto que a amostra abrange maior número de participantes. Observa-se na tabela 6 que os fóruns de dúvidas que geraram maior discussão foram os de “Dúvidas - Módulo 4”, que tratou dos temas de condicionais simples e compostas, e o de “Dúvidas - Trabalho Final”, onde os participantes desenvolveram um jogo utilizando os conceitos estudados.

Tabela 5: N° de interações por fórum 1° oferta.

Tópico	Respostas
Dúvidas - Módulo 1	7
Dúvidas - Módulo 2	3
Dúvidas - Módulo 3	3
Dúvidas - Módulo 4	1
Dúvidas - Módulo 5	6
Dúvidas - Módulo 6	0
Dúvidas - Módulo 7	2

Fonte: os autores

Tabela 6: N° de interações por fórum 2° oferta.

Tópico	Respostas
Dúvidas - Módulo 1	31
Dúvidas - Módulo 2	36
Dúvidas - Módulo 3	26
Dúvidas - Módulo 4	38
Dúvidas - Módulo 5	19
Dúvidas - Trabalho Final	46

Fonte: os autores

4.3 Medida de aprendizado em programação e em Scratch na 1° oferta

No início do projeto foram coletadas algumas informações dos participantes referentes ao nível de conhecimento e domínio nos conceitos de programação e desenvolvimento na plataforma Scratch. Nos questionários pré e pós curso coletou-se a opinião dos próprios participantes, que deram uma nota de 1 a 5 (1 - Muito baixo, até 5 - Muito alto) referente aos dois parâmetros estudados. No apêndice A ilustra-se tal cenário para a primeira oferta do curso, com uma tabela formada pelos valores que representam os 2 parâmetros estudados, sendo analisado somente os resultados daqueles que concluíram mais de 75% das atividades, ou seja, um total de 13 participantes.

Para analisar e comprovar o que os resultados representam foi preciso aplicar os testes estatísticos. Com isso, foi utilizado o teste de Shapiro-Wilk, indicado para amostras com $2 < n < 51$, para verificar a normalidade dos dados (Leotti, 2005). O teste de Shapiro-Wilk é obtido dividindo-se o quadrado da combinação linear apropriada dos valores ordenados da amostra pela variância simétrica estimada.

Apresenta-se no apêndice C os resultados dos testes estatísticos referentes aos dados obtidos (prévios e posteriores) na primeira oferta da unidade instrucional. No teste de Shapiro-Wilk, com um $\alpha = 0,05$, se o valor de $p < \alpha$ então nega-se a hipótese nula que indica para a normalidade dos dados. Portanto, como os valores de p mostrados no quadro 1 do apêndice C são sempre menores que alfa, os dados não apresentam uma distribuição normal, e testes não-paramétricos foram aplicados.

Para isso, foi escolhido o teste de Wilcoxon. Este teste é uma prova não-paramétrica



destinada a comparar dados pareados de uma amostra que foram obtidos antes e depois de algum evento (Neto, 2003). Primeiro, analisou-se os dados relacionados ao nível de conhecimento de Scratch, avaliado pelos próprios participantes antes e após o curso. Neste teste, a hipótese de nulidade sempre aponta que não há diferença significativa entre os valores, enquanto que a hipótese alternativa aponta para a ocorrência de diferença entre os resultados prévios e posteriores. Nível de decisão: $\alpha = 0.01$.

Observa-se no quadro 2 do apêndice C que o valor de $p=0.0015$ é altamente significativo em relação a α , logo rejeita-se a hipótese de nulidade e aceita-se a alternativa. Portanto, como o saldo é positivo, os valores que indicam conhecimento em Scratch são maiores após a aplicação do curso. Isso mostra que, segundo os participantes, o curso auxiliou significativamente no aumento do conhecimento da plataforma Scratch.

Da mesma forma, agora analisando em relação ao nível de conhecimento em programação, observa-se que no quadro 3 do apêndice C como $p=0,0033$ é altamente significativo em relação a α , rejeita-se a hipótese de nulidade e aceita-se a alternativa. Portanto, isso mostra que os valores atribuídos ao conhecimento de programação após a realização do curso são maiores, indicando que houve um aumento significativo na aprendizagem de programação pelos participantes envolvidos na unidade instrucional.

4.4 Medida de aprendizado em programação e em Scratch na 2ª oferta

Na segunda oferta do curso, a mesma análise foi feita para os 55 estudantes que concluíram pelo menos 75% das atividades. De forma semelhante, foram aplicados os questionários prévios e posteriores que identificaram a opinião dos participantes quanto ao seu nível de conhecimento em programação e em Scratch. O apêndice B apresenta a relação completa dos resultados obtidos.

Desta forma, aplicou-se um teste estatístico para analisar os resultados obtidos. Para isso, foi utilizado o teste de Lilliefors com o intuito de identificar se os dados apresentam ou não uma distribuição normal. O teste de Lilliefors é uma adaptação do teste de Kolmogorov-Smirnov, calculando a média e o desvio no próprio conjunto de dados, sem necessidade do cálculo prévio (Torman, 2012). Neste caso, não foi usado o teste de Shapiro-Wilk pois a amostra é referente a um $n = 55$, valor maior do que o indicado por esse teste.

Logo, preparadas as amostras a serem estudadas, definimos as hipóteses para o teste de normalidade de Lilliefors, com um nível de decisão $\alpha = 0,05$. Assim como no teste feito nos dados 1ª oferta, as colunas em sequência representam os mesmos parâmetros da tabela presente no apêndice B, que mostram os dados da 2ª oferta. No quadro 4 do apêndice D observa-se que em todas as colunas o valor de p corresponde a $p < 0.01$, pelo teste de Lilliefors nega-se a hipótese de nulidade. Portanto, os dados apresentam uma distribuição anormal, e testes estatísticos não-paramétricos foram aplicados.

Com essa conclusão, utilizamos novamente o teste de Wilcoxon para analisar o significado dos dados da 2ª oferta. Inicialmente, foi aplicado o teste para as duas primeiras colunas do apêndice B, que representam a indicação do grau de conhecimento da plataforma Scratch. Assim como apresentado no teste anterior, a hipótese de nulidade sempre aponta que não há diferença significativa entre valores, e a hipótese alternativa indica diferença significativa entre os valores estudados. Nível de decisão $\alpha = 0.01$.



Diante do cenário apresentado no quadro 5 do apêndice D, como o valor de $p < 0.0001$ é bastante significativo em relação ao nível de decisão alfa, rejeita-se a hipótese de nulidade. Logo, como os valores que indicam o grau de conhecimento em Scratch pós curso são maiores que os valores indicados no pré curso, houve uma melhora significativa no conhecimento dos participantes em relação a plataforma Scratch.

Em mais uma aplicação do teste, agora tendo vista a evolução no quesito conhecimento de programação, observa-se no quadro 6 do apêndice D que o valor de $p < 0.0001$ indica que devemos rejeitar a hipótese de nulidade. Portanto, os valores que representam o nível de conhecimento em programação após a realização do curso são maiores que os valores indicados no pré curso, mostrando que houve uma melhoria significativa no entendimento dos participantes no conceito de programação.

4.5 Comentários dos participantes a respeito das ofertas da unidade instrucional

No final das atividades, juntamente com os questionários sobre conhecimento em programação, Scratch e lógica, os participantes do curso deixaram seus comentários a respeito dos pontos positivos e negativos que tiveram durante a aplicação da unidade instrucional. Segue na tabela 7 alguns desses comentários que podem servir para uma avaliação geral e para uma possível melhora da unidade em futuras aplicações.

Tabela 7. Comentários dos participantes a respeito da unidade instrucional.

Tópico	Comentários
Pontos fortes da experiência	<p>“Os blocos e a quantidade de conteúdo está perfeito para quem não tem muito tempo. O resumo do conteúdo está condizente com um aluno que não conhece programação e a didática de vocês, as vídeo aulas deram conta do recado.”</p> <p>“Gostei muito de participar do curso, bastante interessante, tutores muito atenciosos.”</p> <p>“O conteúdo do curso foi de alto nível para estimular a motivação e auto conhecimento de soluções e atitudes em diversas situações na área da lógica de programação e para desenvolver habilidades nos problemas do cotidiano”</p> <p>“Amei o curso, acredito que eu tenha aumentado e muito meus conhecimentos sobre iniciação de programação.”</p> <p>“Ótimo curso, completíssimo , com ótimas vídeos aulas com explicações fáceis para quem não sabe do assunto, faria outra vez!”</p> <p>“Adorei o curso. Já tinha conhecimento e realizado formações do Scratch anteriormente, mas esse foi bem didático possibilitando quem não tinha conhecimento de acompanhá-lo.”</p> <p>“Aprendi bastante com este curso e aprimorei minha lógica.”</p> <p>“Foi muito importante esse curso pra que eu pudesse ter contato com o scratch e a partir dele começar a aplicar com meu alunos.”</p> <p>“Gostaria de agradecer imensamente pela oportunidade. O curso é bárbaro! Simples e</p>



	objetivo.” “Esse curso me ajudou muito a realizar um sonho de criar jogos eu agradeço a todos os professores deste curso.”
Pontos fracos da experiência	“Poderia ter mais tempo para as postagens” “Poderia ter um tempo maior para desenvolver o projeto” “Atores e cenários limitados. Desenho ou modificação dos atores bastante complexa” “Adorei, só o tempo para realização das atividades que foram muito apertados.” “Senti falta da participação do restante da turma. Gostaria de ter tido mais troca de experiências” “Acho que dentro das atividades propostas poderiam propor a leitura de materiais mais teóricos sobre os conceitos de pensamento computacional, algoritmos e outros, para complementar a parte prática do curso que já é muito boa.”

Em geral, percebe-se pelos comentários que o que mais atraiu e gerou uma avaliação positiva dos participantes foi a estrutura utilizada na construção da unidade educacional *online*. Os tópicos seguiram uma sequência didática organizada onde os estudantes tiveram facilidade em entender e saber o que deveria ser feito.

Por outro lado, o que chamou mais atenção nas avaliações negativas foi que muitos reclamaram da quantidade de tempo dada para a realização das atividades. Esse ponto pode ser levado em consideração em futuras aplicações da unidade educacional *online* tendo em vista que alguns estudantes podem demorar mais tempo para dominar alguns conceitos computacionais.

5 Considerações Finais

Diante dos novos contextos da relação humano-computador, em que a tecnologia está sendo cada vez mais inserida no cotidiano das pessoas, as discussões e avaliações sobre o envolvimento das tecnologias digitais nas atividades da sociedade são extremamente importantes para que não entremos cegos em um mundo ainda pouco conhecido.

Por meio deste trabalho pode-se estudar e conhecer, diante de pesquisas e aplicações, vários cenários que ao serem debatidos levantam novas reflexões e possibilidades. A partir do desenvolvimento da unidade instrucional online, percebe-se que as plataformas Scratch e Code.org são ótimas alternativas de ferramentas para ensino de programação à estudantes que nunca tiveram contato com o tema.

Estas ferramentas se mostraram potenciais desenvolvedoras das capacidades lúdicas dos estudantes, além de ampliar a autoria e a criatividade dos participantes na construção de seus próprios materiais. Isso mostra que é possível inserir tecnologias digitais nos novos processos de ensino-aprendizagem, uma vez que a multidisciplinaridade é favorecida no uso desses ambientes.

Por fim, pode-se inferir que a partir dos resultados obtidos que as atividades desenvolvidas pelos estudantes auxiliaram significativamente na obtenção de conhecimento do



conceito de programação. Além disso, com os resultados e comentários dos participantes da unidade instrucional concluímos que as ferramentas utilizadas no projeto contribuem, auxiliam e facilitam os processos de ensino-aprendizagem, o que mostra a importância do ensino de programação e softwares de autoria para todos, e não somente para estudantes de computação.

Agradecimentos

O projeto apresentado neste artigo foi financiado pelo órgão CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), uma entidade do governo brasileiro focado no desenvolvimento científico e tecnológico e pelo programa de Iniciação Científica (PIBIC/UNIFAL).



Referências

ALMEIDA, Maria Elizabeth Biancocini de. (2001). *Informática e formação de professores*. Brasília: Ministério da Educação/Proinfo.

ARTIFON, Mersio José; ROSSAL, Maria Margareth Lins. (1999). *SISTEMAS DE AUTORIA*.

BITTENCOURT, Ig Bittencourt et al. (2007). *Sistemas de Autoria para Construção de Ambientes Interativos de Aprendizagem Baseada em Agentes*. *Revista Brasileira de Informática na Educação*, v. 15, n. 1.

CASTELLS, Manuel. (2009). *Comunicación y poder*. Madrid: Alianza Editorial.

DA CRUZ ALVES, Nathalia et al. (2016). *Ensino de Computação de Forma Multidisciplinar em Disciplinas de História no Ensino Fundamental--Um Estudo de Caso*. *Revista Brasileira de Informática na Educação*, v. 24, n. 3.

DIAS, Lia Ribeiro. (2011). *Inclusão digital como fator de inclusão social*. *InclusAo digital*.

FINIZOLA, Antonio Braz et al. (2014). *O ensino de programação para dispositivos móveis utilizando o MIT-App Inventor com alunos do ensino médio*. In: *Anais do Workshop de Informática na Escola*. p. 337.

FREIRE, P. (1996). *Pedagogia da Autonomia: saberes necessários a prática educativa*. São Paulo. Paz e Terra.

LEOTTI, Vanessa Bielefeldt; BIRCK, Alan Rodrigues; RIBOLDI, João. (2005). *Comparação dos Testes de Aderência à Normalidade Kolmogorov Smirnov, Anderson-Darling, Cramer-Von Mises e Shapiro-Wilk por Simulação*. *Simpósio de Estatística Aplicada à Experimentação Agrônômica*, v. 11, p. 192.

MASCHIO, Eleandro; DIRENE, Alexandre Ibrahim. (2015). *Múltiplas Representações Externas no Suporte à Aquisição de Conhecimento em Programação de Computadores*. *Revista Brasileira de Informática na Educação*, v. 23, n. 3.

MERCADO, Luis Paulo Leopoldo. (Org.). (2002). *Novas tecnologias na educação: reflexões sobre a prática*. Maceió. Edufal.

MORAES, Raquel de Almeida. (2000). *Informática na educação*. Rio de Janeiro: DPA.

MORAN, José Manuel. (2000). *Ensino e aprendizagem inovadores com tecnologias*. *Informática na educação: teoria & prática*, v. 3, n. 1.



NETO, Arthur Alexandre Hackbarth; STEIN, Carlos Efrain. (2003). Uma abordagem dos testes não-paramétricos com utilização do Excel. I Uma abordagem dos testes não-paramétricos com utilização do Excel.

PEREIRA, Andréia Regina; DE DEUS LOPES, Roseli. (2005). Legal: Ambiente de Autoria para Educação Infantil apoiada em Meios Eletrônicos Interativos. In: Anais do Simpósio Brasileiro de Informática na Educação. p. 23-30.

PINTO, Ivete Martins; BOTELHO, Silvia Silva da Costa. (2012). Ambientes Tecnológicos Lúdicos de Autoria (ATLA): criando espaços de ensino e aprendizagem. Revista Brasileira de Informática na Educação.

POY, R.; GONZALES-AGUILAR, A. (2014). Factores de éxito de los MOOC: algunas consideraciones críticas. Revista Ibérica de Sistemas e Tecnologia da Informação, n. E1, v. 03, p. 95-118.

RAMOS, José Luís; ESPADEIRO, Rui Gonçalo. (2014). Os futuros professores e os professores do futuro. Os desafios da introdução ao pensamento computacional na escola, no currículo e na aprendizagem. Educação, Formação & Tecnologias-ISSN 1646-933X, v. 7, n. 2, p. 4-25.

SCAICO, P. D.; LIMA, A.A.; SILVA, J. B. B.; AZEVEDO, S. ; PAIVA, L. F.; RAPOSO, E. H.; ALENCAR, Y.; MENDES, J. P.; SCAICO, A. (2012). Ensino de Programação no Ensino Médio: Uma Abordagem Orientada ao Design com a linguagem Scratch. Revista Brasileira de Informática na Educação, Volume 21.

SERRÃO, Tássia et al. (2016). CONSTRUÇÃO AUTOMÁTICA DE REDES SOCIAIS MÓVEIS NO AMBIENTE MOODLE. Revista Brasileira de Informática na Educação, v. 24, n. 1.

SILVEIRA, Sérgio Amadeu. A noção de exclusão digital diante das exigências de uma cibercidadania. In: HETKOWSKI, Tânia Maria (Org.). (2008). Políticas públicas & inclusão digital. Salvador: EDUFBA.

SILVEIRA, Sérgio Amadeu. (2011). Para além da inclusão digital: poder comunicacional e novas assimetrias. Inclusão digital.

TORMAN, Vanessa Bielefeldt Leotti; COSTER, Rodrigo; RIBOLDI, João. (2012). Normalidade de variáveis: métodos de verificação e comparação de alguns testes não-paramétricos por simulação. Clinical & Biomedical Research, v. 32, n. 2.

VALENTE, J. M. (2013). Diferentes usos do computador na Educação. Universidade Estadual do Paraná. Disponível em <http://ffalm.br/gied/site/artigos/diferentesusoscomputador.pdf>



VON WANGENHEIM, Christiane Gresse; NUNES, Vinícius Rodrigues; DOS SANTOS, Giovane Daniel. (2014). Ensino de Computação com SCRATCH no Ensino Fundamental–Um Estudo de Caso. Revista Brasileira de Informática na Educação, v. 22, n. 03, p. 115.

Apêndice

A. Notas dos questionários pré e pós curso e avaliação dos alunos sobre grau de conhecimento 1^o oferta

Participantes	Conhecimento Scratch (pré-curso)	Conhecimento Scratch (pós-curso)	Saldo	Conhecimento programação (pré-curso)	Conhecimento programação (pós-curso)	Saldo
1	1	5	4	3	3	0
2	1	5	4	1	5	4
3	1	5	4	1	4	3
4	1	5	4	3	5	2
5	4	5	1	5	5	0
6	3	5	2	2	4	2
7	1	5	4	1	5	4
8	1	5	4	2	5	3
9	1	4	3	1	3	2
10	1	4	3	2	4	2
11	1	5	4	3	5	2
12	1	5	4	1	5	4
13	1	5	4	1	3	2

Fonte: os autores



B. Notas dos questionários pré e pós curso e avaliação dos alunos sobre grau de conhecimento 2º oferta

Participantes	Conhecimento Scracth (pré-curso)	Conhecimento Scracth (pós-curso)	Saldo	Conhecimento programação (pré-curso)	Conhecimento programação (pós-curso)	Saldo
1	2	3	1	4	5	1
2	2	3	1	3	4	1
3	3	4	1	3	5	2
4	2	3	1	2	4	2
5	1	3	2	1	4	3
6	2	4	2	3	4	1
7	3	3	0	2	3	1
8	3	3	0	3	5	2
9	1	3	2	1	3	2
10	3	4	1	1	3	2
11	1	4	3	1	4	3
12	2	4	2	3	4	1
13	3	5	2	2	5	3
14	3	3	0	4	4	0
15	2	3	1	1	3	2
16	4	5	1	3	5	2
17	1	2	1	1	2	1
18	2	5	3	4	5	1
19	1	3	2	4	4	0
20	3	5	2	2	5	3
21	3	4	1	3	4	1
22	5	5	0	3	4	1
23	1	3	2	2	3	1



24	2	3	1	2	4	2
25	3	4	1	1	4	3
26	2	4	2	3	4	1
27	3	3	0	2	3	1
28	3	3	0	4	5	1
29	3	3	0	3	3	0
30	3	3	0	3	4	1
31	1	2	1	1	3	2
32	5	5	0	2	5	3
33	4	5	1	2	5	3
34	1	3	2	1	3	2
35	2	3	1	2	3	1
36	4	5	1	5	5	0
37	1	2	1	1	2	1
38	2	4	2	3	5	2
39	3	3	0	4	5	1
40	3	4	1	4	4	0
41	3	5	2	3	5	2
42	1	4	3	1	4	3
43	1	2	1	1	2	1
44	2	4	2	1	5	4
45	1	4	3	1	5	4
46	1	1	0	1	3	2
47	1	1	0	1	3	2
48	1	3	2	1	3	2
49	2	4	2	1	5	4
50	3	5	2	3	5	2
51	3	4	1	4	4	0



52	1	3	2	1	4	3
53	2	5	3	1	5	4
54	3	4	1	3	4	1
55	5	5	0	2	5	3

C. Resultados estatísticos referentes a 1º oferta

Quadro 1. Resultados teste de normalidade Shapiro-Wilk

Resultados	- 1 -	- 2 -	- 3 -	- 4 -
Tamanho da amostra =	13	13	13	13
Média =	1.3846	4.8462	2.0000	4.3077
Desvio padrão =	0.9608	0.3755	1.2247	0.8549
W =	0.4663	0.4456	0.8006	0.7437
p =	0.0057	0.0055	0.0098	0.0091

Fonte: Software BioEstat 5.0

Quadro 2. Resultados do teste de Wilcoxon em relação ao conhecimento de Scratch

	Resultados Cols. 1 e 2
T =	0
Número de pares =	13
Z =	3.1798
p-valor (unilateral) =	0.0007
p-valor (bilateral) =	0.0015

Fonte: Software BioEstat 5.0

Quadro 3. Resultados do teste de Wilcoxon em relação ao conhecimento em programação

	Resultados Cols. 3 e 4
T =	0
Número de pares =	11
Z =	2.9341
p-valor (unilateral) =	0.0017
p-valor (bilateral) =	0.0033

Fonte: Software BioEstat 5.0



D. Resultados estatísticos referentes a 2º oferta

Quadro 4. Resultados do teste de normalidade de Lilliefors

	- 1 -	- 2 -	- 3 -	- 4 -
Tamanho da amostra =	55	55	55	55
Desvio máximo =	0.1800	0.2047	0.2243	0.2226
Valor crítico (0.05) =	0.1195	0.1195	0.1195	0.1195
Valor crítico (0.01) =	0.1390	0.1390	0.1390	0.1390
p(valor)	< 0.01	< 0.01	< 0.01	< 0.01

Fonte: Software BioEstat 5.0

Quadro 5. Resultados do teste de Wilcoxon em relação ao conhecimento de Scratch

	Resultados Cols. 1 e 2
T =	21
Número de pares =	42
Z =	5.3828
p-valor (unilateral) =	<0.0001
p-valor (unilateral) =	<0.0001

Fonte: Software BioEstat 5.0

Quadro 6. Resultados do teste de Wilcoxon em relação ao conhecimento de programação

	Resultados Cols. 3 e 4
T =	0
Número de pares =	49
Z =	6.0927
p-valor (unilateral) =	<0.0001
p-valor (unilateral) =	<0.0001

Fonte: Software BioEstat 5.0

E. Mapeamento prévio: Softwares de autoria

Nome	Acesso (pago ou livre)	Público-alvo	Tipo de mídia	Execução do produto
Scratch MIT	Livre	Acima de 4 anos	animação/app	web/desktop
Codeorg	Livre	Acima de 4 anos	animação	web
Ardora	Livre	Adultos/Jovens	recursos web	desktop
CourseLab	Livre	Professores	curso e-learning	desktop
MyUdutu	Livre	Professores	curso e-learning	web



Xerte	Livre	Professores	curso e-learning	web/desktop
HagáQuê (HQ)	Livre	Acima de 4 anos	animação	desktop
KompoZer	Livre	Adultos/Jovens	páginas web	desktop
Mblock	Livre	Adultos/Jovens	Prog Arduino	desktop
Scratch4A	Livre	Adultos/Jovens	Prog Arduino	desktop
AppInventor	Livre	Adultos/Jovens	Apps celular	web
Kturtle	Livre	Acima de 10 anos	animação/app	desktop
Hot Potatoes	Livre	Professores	mat. e-learning	web
Yenka	Livre	Professores	mat. e-learning	desktop
RoboMind	Livre	Acima de 9 anos	Prog Robô Lego	Robô Lego
GameMaker	Livre	Adultos/Jovens	Jogo Arcade	desktop/celular
GameSalad	Livre	Adultos/Jovens	Jogo Arcade	web/celular
Macromedia Flash MX	Livre	Adultos/Jovens	Anim/app/jogo	web
GreenFoot	Livre	Acima de 14 anos	Anim/app/jogo	desktop
BlueJ	Livre	Acima de 14 anos	aplicação	desktop
VisualClass	Pago	Professores	mat. e-learning	desktop
Everest	Pago	Acima de 9 anos	animação/mat.	desktop
Imagine	Pago	Acima de 9 anos	animação/mat.	desktop
CodeCombat	Livre	Acima de 4 anos	animação	web
LightBot	Pago	Acima de 4 anos	animação	web/desk/cel
Alice	Livre	Jovens	Anim/app/jogo	desktop
Kid Pix	Pago	Acima de 4 anos	Animações	desktop
Asymetrix Toolbook	Livre	Professores	App/e-learning	web/desktop
Blocky Games	Livre	Acima de 7 anos	Animação	web
OzoBlockly	Livre	Acima de 10 anos	Animação	web
Micro:bit	Livre	Acima de 7 anos	Animação/app	web/desktop
BlocklyProp	Livre	Jovens	Anim/Prog Robo	web/robo
Wonder	Pago	Acima de 7 anos	Prog Robo/apps	web/robo