

Aplicação da Taxonomia de Bloom no ensino de programação com Scratch

Ana Liz Souto Oliveira de Araújo, Pasqueline Dantas Scaico, Luiz Fernando de Paiva, Hévellyn de Moraes Rabêlo, Luan de Luna Santos, Francisco Ittalo Ribeiro Pessoa, Jonas Mendonça Targino, Leonardo dos Santos Costa

Centro de Ciências Aplicadas e Educação – Universidade Federal da Paraíba (UFPB) -
Rua da Mangueira, s/n - CEP 58.297-000 – Rio Tinto – PB – Brasil

{analiz, pasqueline, fernando.paiva, hevellyn.morais, luan.luna, francisco.itallo, jonas.mendonca, leonardo.costa}@dce.ufpb.br

***Abstract.** Teaching how to program in schools is an initiative growing and there are many challenges to be overcome. The main challenges involve the selection of an appropriate language and a tool to guide the learning. In this context, this paper presents the results of artifacts and experience in a programming course using Scratch as a language and Bloom's taxonomy in educational planning. The results show the performance of students and the main difficulties presented.*

***Resumo.** Ensinar programação em nível médio é uma iniciativa em crescimento e ainda com desafios a serem superados. Os principais desafios envolvem a escolha de uma linguagem apropriada para o público alvo e um instrumento que oriente no ensino e na avaliação das atividades realizadas. Neste âmbito, este artigo apresenta os artefatos gerados e os resultados da experiência de um curso de introdução à programação utilizando a linguagem de programação Scratch e empregando a taxonomia de Bloom no planejamento didático. Os resultados obtidos detalham o desempenho dos alunos e as principais dificuldades apontadas por eles.*

1. Introdução

Aprender programação é uma atividade importante, principalmente por desenvolver um conjunto de habilidades úteis em diversas áreas do conhecimento. Todavia, seu propósito geral não deve estar apenas focado na produção final de código. O cerne desta atividade envolve, sobretudo, desenvolver nos alunos competências relacionadas à abstração, raciocínio lógico, decomposição de problemas e criação de estratégias para a produção de soluções. Esses conceitos estão ligados diretamente aos aspectos cognitivos e psicológicos que precisam ser despertados, fortalecidos e praticados [Sudol, 2011].

O ensino de programação no nível médio vem se mostrando uma iniciativa promissora, por se entender que esse conjunto de habilidades desenvolvidas pode ser empregado no cotidiano, não estando restrito apenas aos profissionais de computação. Uma das principais dificuldades para apoiar essa iniciativa encontra-se na escolha de uma linguagem adequada a esse público, que desperte a atenção e facilite a aprendizagem. Os trabalhos de [Aureliano e Tedesco, 2012], [Pereira *et al*, 2012] e [Scaico *et al*, 2012] apontam que o *Scratch* é uma linguagem adequada para a realidade das escolas brasileiras por utilizar um conceito inovador de desenvolvimento orientado

ao design, que privilegia a computação criativa, permite a manipulação de mídias, para a criação de animações e jogos, e está disponível gratuitamente no idioma português.

Outro desafio encontrado para se ensinar programação consiste em escolher um instrumento adequado que possa nortear o ensino e a avaliação de aprendizagem. Afirma-se que há uma enorme dificuldade por parte dos professores em mensurar objetivamente a aprendizagem dos alunos nas disciplinas de programação [Jesus e Raabe, 2009]. Para auxiliar nesse quesito, a taxonomia de Bloom [Bloom, 1956] [Krathwohl, 2002] é apontada como instrumento aplicável também no ensino de programação [Aureliano e Tedesco, 2012].

A taxonomia de Bloom classifica os domínios cognitivos em seis categorias principais: conhecimento, compreensão, aplicação, análise, síntese e avaliação. Essas categorias são dispostas em níveis de complexidade, da mais elementar para a mais complexa. Dessa forma, ela pode servir como um recurso de referência para orientar o ensino a partir de uma hierarquia crescente de habilidades cognitivas. Mesmo com essas características, [Jesus e Raabe, 2009] afirmam que ainda existe uma dificuldade de interpretar e aplicar as categorias da taxonomia no contexto dos exercícios e práticas em programação.

Diante do cenário apresentado, este trabalho apresenta os artefatos gerados e os resultados da experiência em um curso de introdução à programação para alunos do ensino médio de uma escola pública. Um diferencial deste trabalho está na escolha da taxonomia de Bloom como instrumento empregado para o ensino de programação com a linguagem *Scratch*. Uma das contribuições do trabalho consiste na elaboração e disponibilização do plano de curso, planos de aulas e cadernos de atividades elaborados segundo a taxonomia de Bloom. Esse material pedagógico está disponível gratuitamente no site do projeto <http://www2.ccae.ufpb.br/pibid/?page_id=15>. Portanto, o curso pode ser aplicado em outros ambientes.

Este trabalho encontra-se organizado da seguinte forma: a seção 2 descreve a metodologia utilizada. A seção 3 explica a taxonomia de Bloom aplicada ao contexto do curso de *Scratch*. A seção 4 apresenta as atividades criadas e executadas no curso como forma de avaliar a aprendizagem. A seção 5 relata as dificuldades e os resultados atingidos pelos alunos no curso. A seção 6 apresenta as reflexões e as considerações finais do trabalho.

2. Metodologia de trabalho

O curso foi elaborado por estudantes e professores da Licenciatura em Ciência da Computação da Universidade Federal da Paraíba, que ao longo dos últimos dois anos têm conduzido ações voltadas para a introdução do pensamento computacional e ensino de programação no cotidiano de jovens, alunos do ensino médio de escolas públicas da região [Scaico et al, 2012]. Assim, este relato descreve a continuação de um trabalho que tem procurado estabelecer uma metodologia de ensino de programação que combine elementos de *Design thinking* e a taxonomia de Bloom na tentativa de estabelecer a aproximação de jovens com a educação em Computação.

A metodologia de trabalho, desde sempre utilizada, preserva os princípios do poder de concepção do estudante, que o expõe a situações que exigem a produção de novas ideias a partir de um processo criativo, e não apenas a utilização (ou interação) de ideias previamente criadas por outras pessoas; a personalização dos elementos que serão

construídos, na qual o estudante cria algo que é pessoalmente significativo e relevante; a necessidade de colaboração, que se manifesta através do trabalho em equipe e da discussão das ideias e soluções, e por fim, o poder da reflexão, no qual o estudante é estimulado a rever e repensar as práticas criativas na elaboração dos seus algoritmos.

Para estabelecer a experiência de ensinar programação para jovens com pouco contato com Computação foi realizado um estudo no intuito de conhecer e selecionar uma linguagem de programação adequada ao perfil do público existente. Neste caso, a linguagem adotada foi a *Scratch*. Buscou-se também melhorar as estratégias orientadoras do processo de ensino, de maneira que a construção dos saberes pudesse ser construída gradualmente, caracterizando, portanto, um processo engajador, significativo e estimulante para o aluno. A base teórica que vem sendo utilizada para nortear este aspecto da metodologia é justamente a taxonomia de Bloom.

Nesta edição, o curso foi planejado com uma carga horária de 16 horas distribuídas em 04 aulas de 4 horas cada. Realizou-se ainda um quinto encontro, chamado de *Scratch Day*, com o objetivo de divulgar os programas que foram produzidos pelos alunos, apresentar informações sobre cursos na área de computação e entregar certificados de participação. O curso foi executado duas vezes com alunos do ensino médio de escola pública. Na primeira edição teve-se uma média de 23 alunos e na segunda, uma média de 40 alunos. Todas as atividades dos cursos foram realizadas nos laboratórios da Universidade Federal da Paraíba, na cidade de Rio Tinto - PB.

3. O uso da taxonomia de Bloom no curso de *Scratch*

Scratch é uma linguagem de programação visual desenvolvida pelo *Lifelong Kindergarten Group* [Lifelong Kindergarten, 2007], grupo de pesquisa do MIT Media Lab, em 2007, visando auxiliar o aprendizado de conceitos computacionais e matemáticos por pessoas maiores de 8 anos, e que não possuam experiência em programação [Scratch, 2011]. Ademais, é uma linguagem cujo paradigma é orientação a objetos e a eventos, além de ser um software educacional, com tipagem dinâmica e de código aberto.

Um fator motivacional para escolha dessa linguagem foi sua interface gráfica, proporcionando a criação de programas por meio de blocos de montar. Segundo [Scaico *et al*, 2012] isso facilita o aprendizado do aluno, tendo em vista que em outras linguagens haveria uma preocupação a mais: a sintaxe. Outra vantagem é existir uma versão gratuita em português para seu uso. Uma vez observado que *Scratch* é uma linguagem adequada ao contexto de programação no ensino médio, viu-se a necessidade de um instrumento de ensino no intuito de se aplicar um critério pedagógico que possibilite obter uma boa resposta da aprendizagem. Assim, o curso de *Scratch* teve seu planejamento didático-pedagógico, como também os instrumentos de avaliação, apoiados pela taxonomia de Bloom [Bloom 1956].

A taxonomia de Bloom consiste em um instrumento cujo objetivo é apoiar a identificação de uma estrutura hierárquica com objetivos educacionais bem definidos, do mais simples para o mais complexo, abrangendo seis principais categorias de domínio cognitivo: conhecimento, compreensão, aplicação, análise, síntese e avaliação. Aliada a definição de cada um desses níveis do domínio cognitivo há um conjunto de verbos associados sugeridos, os quais auxiliam na elaboração de questões de avaliação para serem aplicados em cada uma das categorias da taxonomia. Assim, para que o

aluno adquira as habilidades do próximo nível, é necessário que ele tenha adquirido habilidades do nível anterior.

Tendo em vista a possibilidade do uso da linguagem *Scratch* e da taxonomia de Bloom como instrumento de ensino e avaliação, o planejamento do curso foi construído, de forma que cada aula e exercício fossem executados aplicando as categorias do domínio cognitivo descritas. Cada aula seguiu uma ordem de complexidade já proposta pela própria taxonomia em questão. Cada atividade empregou um dos verbos da lista definida pela taxonomia e relacionada ao seu nível, que, por conseguinte, este nível esteve relacionado à aula em que foi trabalhado. O Quadro 1 mostra os níveis da taxonomia, uma breve descrição e as sugestões de verbos para aplicação nas atividades.

Quadro 1: Divisões dos níveis da taxonomia de Bloom

	Nível	Definição	Amostra de Verbos
1	Conhecimento	O aluno irá recordar ou reconhecer informações, ideias e princípios na forma em que foram aprendidos.	Escreva, Liste, Rotule, Nomeie, Diga e Defina.
2	Compreensão	O aluno compreende ou interpreta informação com base em conhecimento prévio ou novo.	Explique, Resuma, Parafraseie, Descreva.
3	Aplicação	O aluno seleciona, transfere e usa dados e princípios para completar um problema ou tarefa com um mínimo de supervisão.	Use, Compute, Resolva, Demonstre, Aplique, Construa.
4	Análise	O aluno distingue, classifica e relaciona pressupostos, hipóteses, evidências ou estruturas de uma declaração ou questão.	Analise, Categorize, Compare, Contraste e Separe.
5	Síntese	O aluno cria, integra e combinam ideias em um produto, plano ou proposta, novos para ele.	Crie, Planeje, Elabore hipótese(s), Invente e Desenvolva.
6	Avaliação	O aluno aprecia, avalia ou critica com base em padrões e critérios específicos.	Julgue, Recomende, Critique e Justifique.

Portanto, cada aula foi norteada pedagogicamente por um ou mais níveis da taxonomia e empregou os verbos sugeridos para orientar o processo avaliativo de aprendizagem. A seguir, são apresentados detalhamentos da aplicação efetiva das categorias do domínio cognitivo moldado ao conteúdo de programação ministrado.

3.1 Aplicação da taxonomia de Bloom nos conteúdos

Os conteúdos de introdução a algoritmos, linguagem de programação, apresentação do ambiente *Scratch* e plano cartesiano (conceito que orienta a movimentação dos elementos no *Scratch*) foram ministrados norteados pelos níveis conhecimento e compreensão da taxonomia. O nível conhecimento estabelece que lembrar consiste em reconhecer e recordar informações já adquiridas e guardadas na memória de longa duração. Já o nível compreensão afirma que entender consiste na capacidade de fazer sua própria interpretação do material educacional.

Para alcançar as metas propostas pela taxonomia, o aluno foi provocado a recordar de exemplos do seu cotidiano, como ir à escola, trocar uma lâmpada, fazer uma ligação de celular e, nesse contexto, foi introduzido os conceitos introdutórios de

algoritmo. Em seguida, os alunos foram apresentados aos recursos do ambiente *Scratch* de maneira onde eles pudessem conhecer e compreender como utilizá-lo. Mais uma vez, eles foram provocados a recordar conhecimentos previamente aprendidos, agora matemáticos, sobre o plano cartesiano, para compreender como acontece a movimentação de elementos no ambiente *Scratch*.

O conteúdo de laço de repetição foi apresentado aos alunos e empregaram-se os níveis de aplicação e análise para avaliar a aprendizagem. O terceiro nível da taxonomia, aplicação, refere-se a usar procedimentos e conteúdos aprendidos em uma situação familiar ou nova. Assim, foi proposto um exercício onde um garoto precisava percorrer um o campo de *baseball*. Nesse exercício, os alunos aplicaram seu entendimento sobre posição, movimentação, aparência e controle, conceitos importantes existentes dentro do ambiente de codificação do *Scratch*. Essa atividade visou analisar a capacidade dos alunos de realizarem abstrações e uso de laço de repetição.

O quarto nível da taxonomia, análise, consiste em dividir o conhecimento em partes e pensar como essas partes se relacionam com a estrutura geral. Então, foi sugerida uma atividade onde os alunos desenvolveram um diálogo entre dois *sprites* (objetos que realizam ações em um projeto no *Scratch*) os quais interagiriam entre si de forma consecutiva utilizando estruturas condicionais “SE” e “SE NÃO”. Os alunos foram avaliados se conseguiram analisar que cada *sprite* precisava utilizar bloco de comandos e atribuições diferentes, e se conseguiram organizar no tempo e no espaço, de forma que a animação apresentasse uma coerência lógica de acontecimentos.

O quinto nível da taxonomia, síntese, refere-se à capacidade de planejamento e criação a partir do conteúdo ensinado. Assim, foi proposta uma prática onde os alunos tiveram que, primeiramente, planejar como realizariam a atividade e, posteriormente, implementá-la no *Scratch*. Dessa forma, eles deveriam ser capazes de reunir os elementos de programação ensinados e desenvolver o programa planejado.

No intuito de avaliar se os alunos tinham aprendido os conteúdos, baseado no sexto nível da taxonomia, foi proposta uma atividade de depuração de código. Assim, foi entregue aos alunos um código em *Scratch* que continha erros. Eles precisavam julgar o que estava certo, encontrar os erros e corrigi-los. Em uma última atividade foi sugerido que os alunos elaborassem uma animação no *Scratch* usando sua imaginação e empregassem conceitos aprendidos no curso. O objetivo dessa atividade foi fazer com que os alunos trabalhassem sozinhos, e utilizando de sua criatividade, desenvolvessem a habilidade de julgar se: (i) o que estavam fazendo era coerente, (ii) encaixa nos requisitos da atividade e (iii) aprenderam os conceitos de programação e elementos da linguagem *Scratch*.

4. Caderno de atividades

Os conceitos da taxonomia de Bloom foram aplicados no curso de *Scratch* tanto nas aulas teóricas, como na elaboração de um caderno de atividades. Em cada aula foi aplicada uma atividade relacionada com o conteúdo ministrado. Cada quesito das atividades emprega os verbos de ação e substantivos sugeridos pela taxonomia que procuram descrever as metas de aprendizagens desejadas.

As seções seguintes apresentam as atividades, os quesitos que os alunos apresentaram maior dificuldade e a justificativa encontrada para tal resultado. Devido à limitação de espaço físico, apenas parte das atividades serão exibidas na íntegra.

4.1. Atividade 1

Na primeira atividade, os alunos precisavam definir e explicar conceitos de algoritmos, linguagens de programação e também conceitos relacionados à ferramenta do *Scratch*. O Quadro 2 mostra parte dessa atividade, dando destaque aos verbos sugeridos nos níveis conhecimento e compreensão da taxonomia.

Quadro 2: Quesitos da Atividade 1

Quesito 1 - Defina o que é algoritmo com suas palavras. Diga um exemplo.
Quesito 2 - Explique o que é linguagem de programação com suas palavras.
Quesito 3 - Resuma com suas palavras: o que é um <i>sprite</i> ?
Quesito 4 - Liste três exemplos de comandos de movimento.
Quesito 5 - Liste três exemplos de comandos de controle.
Quesito 6 - Liste três exemplos de comandos de aparência.

Nesta atividade, observou-se maior índice de erro nas respostas dos quesitos 2 e 3. Considerou-se que a dificuldade com o quesito 2 ocorreu devido ao fato que era a primeira vez que os alunos tinham conhecimento que os programas eram criados por meio de uma linguagem de programação, ou seja, não existia a percepção que um programa é construído por meio de um código cujo o computador é capaz de interpretar. Já no quesito 3, a dificuldade inicial era de definir com palavras o significado de *sprite*, mas na prática, os alunos conseguiam perceber sua utilidade.

4.2. Atividade 2

A segunda atividade foi orientada pelas categorias de aplicação e análise da taxonomia. Os exercícios foram propostos para que os alunos tentassem identificar quais blocos de comandos foram usados na animação que foi apresentada. Também foi solicitado que eles separassem e aplicassem os comandos ensinados, e descrevessem quais as diferenças entre os comandos de repetição. O Quadro 3 mostra parte da atividade aplicada.

Quadro 3: Quesitos da Atividade 2

Quesito 1 - Utilizando os blocos de comandos citados acima no caderno, utilize os comandos (Movimento, Controle e Aparência) para construir o projeto especificado na folha em anexo.
Quesito 2 - Separe quais comandos foram utilizados para a elaboração do projeto.
Quesito 3 - Aplique mais algum(s) comando(s) para incrementar a animação.
Quesito 4 - Analise o que aconteceria na animação se, ao invés de utilizar o comando “Sempre”, fosse empregado o comando “Repita”. Explique em síntese.
Quesito 5 - Demonstre por meio de palavras, dentro da estrutura da animação, o que acontece dentro do comando “Repita”.

Nesta atividade, os alunos apresentaram maior dificuldade nos quesitos 2 e 3. No quesito 2, considerou-se que ocorreu porque parte dos alunos não conseguiam distinguir os comandos utilizados, consideravam apenas a animação criada. Já no quesito 3, observou-se que o item não foi respondido devido a um bloqueio de se aplicar criatividade e abstração utilizando os elementos da linguagem.

4.3. Atividade 3

Na terceira atividade foi apresentado um exercício onde o aluno deveria exercitar a capacidade de síntese, fazendo relação ao quinto nível da taxonomia de Bloom. O aluno usaria da criatividade para criar uma história, identificar quais comandos seriam necessários para codificá-la e desenvolvê-la em *Scratch*. Nesta atividade, 90% dos alunos conseguiram concluí-la com sucesso. Em casos isolados, observou-se que não se conseguiu concluir a atividade na totalidade porque não houve um planejamento adequado na história, ou porque foi feita muitas alterações durante a execução da atividade. Nesses casos, a atividade ficou incompleta.

4.4. Atividade 4

Na quarta atividade foi proposto mais um exercício de criação, dessa vez explorando o quinto e sexto nível da taxonomia, síntese e avaliação, respectivamente. O aluno usaria sua criatividade para criar uma animação no *Scratch*, usando o maior número de comandos possível. Além de exercitar a capacidade de elaboração, foi explicado que, naquele momento, eles estavam avaliando sua própria capacidade de abstração para criar algoritmos usando uma linguagem de programação. Desta vez todos os alunos conseguiram concluir a atividade proposta.

5. Resultados

Ao final dos cursos foi aplicado um questionário avaliativo com intuito de analisar quais assuntos e aulas ministradas os alunos sentiram maior dificuldade e maior facilidade para aprender. Os gráficos mostrados nesta seção apresentam os resultados das duas edições do curso.

A Figura 1 apresenta um gráfico com cinco tópicos nos quais os alunos sentiram maior dificuldade levando em consideração questões gerais de programação. Eles puderam escolher mais de um item nessa pesquisa. Assim, pode-se observar que os maiores obstáculos apresentados, em ordem decrescente de porcentagem, foram (i) abstração, (ii) planejamento do projeto, (iii) entendimento do passo a passo do código, (iv) raciocínio lógico e por último (iv) iniciar do projeto.

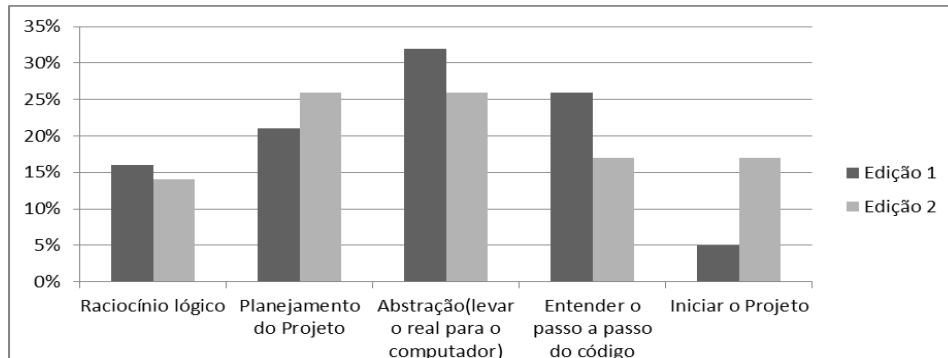


Figura 1: Principais dificuldades apresentadas pelos alunos

A Figura 2 mostra as dificuldades apresentadas pelos alunos ao se adotar como critério o conteúdo ministrado em cada aula. Observa-se que a aula 4, que explorou o assunto operadores lógicos e variáveis, apresentou maior incidência de respostas, porque esses assuntos não foram bem relacionados com os conteúdos correlatos no ensino médio. Já na aula 3, que explorou o assunto som, sensores e canetas, considerou-se que a dificuldade existiu devido os alunos não estarem habituados com a abstração da utilização de sensores e canetas no ambiente *Scratch*. Já na aula 1, percebeu-se que a dificuldade decorreu devido aos alunos nunca terem se deparado com os conceitos de algoritmo e linguagem de programação.

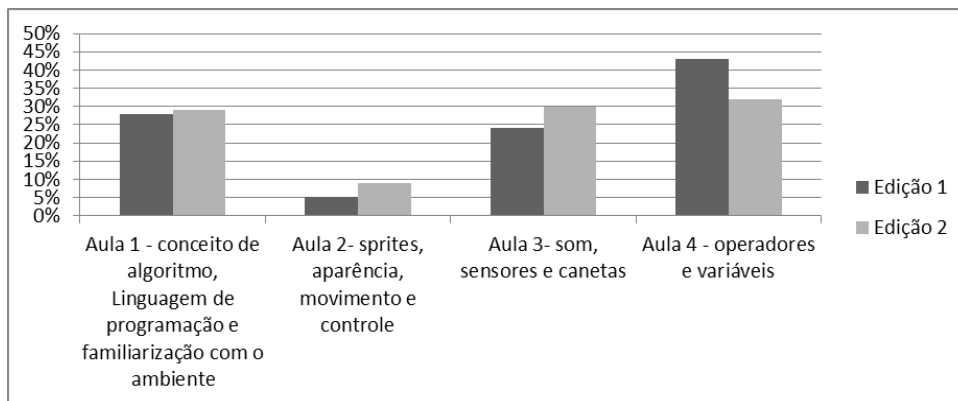


Figura 2: Principais dificuldades agrupadas por aula

A Figura 3 ilustra que os alunos tiveram maior facilidade em aprender o conteúdo da aula 2, referente a utilização de *sprites*, aparência, movimento e controle, conceitos essenciais existentes dentro do ambiente de codificação do *Scratch*. Este fenômeno pode ter ocorrido em função de estes serem conceitos fáceis de serem demonstrados de maneira visual, como por exemplo, demonstrar um objeto se movimentando na tela.

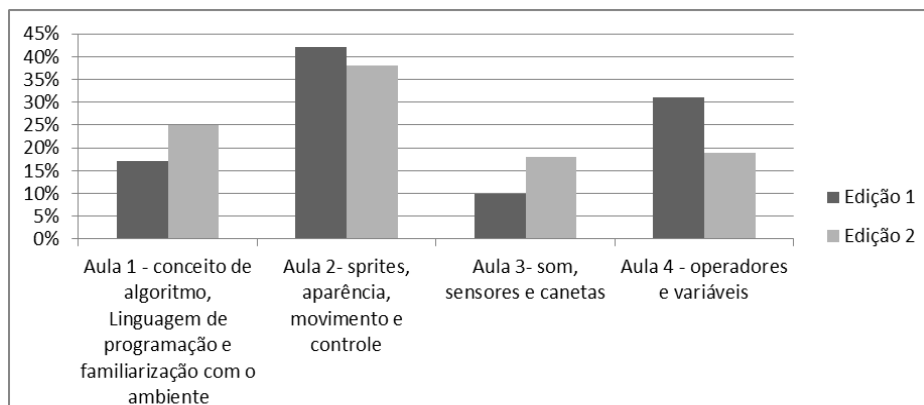


Figura 3: Facilidades identificada por aula

Para finalizar a avaliação do curso, uma última pergunta feita aos alunos indagava se eles tinham interesse em continuar a aprender programação com outras linguagens. Para surpresa da equipe organizadora, todos os alunos responderam que gostariam de aprender outra linguagem.

6. Considerações finais

Esse trabalho apresentou os artefatos gerados e os resultados de uma experiência com a aplicação da taxonomia de Bloom como estratégia pedagógica para o processo de ensino-aprendizagem de programação com a linguagem *Scratch*. Pode-se afirmar que o trabalho mostrou resultados proveitosos, tanto para os alunos do ensino médio quanto para equipe organizadora.

Considera-se que a experiência que os discentes do ensino médio tiveram com a prática de programar tenha contribuído para desenvolver habilidades de abstração, raciocínio lógico e criatividade. Alguns alunos apresentaram alta desenvoltura em aplicar os conceitos ensinados, se destacando pelos programas originais gerados. Ademais, espera-se que a vivência possa ter fornecido uma maior clareza sobre programação e o profissional da computação, influenciando-os futuramente de forma positiva na escolha de um curso técnico ou superior.

A taxonomia de Bloom teve papel fundamental como instrumento para a análise e avaliação do ensino-aprendizagem dos alunos nos cursos ministrados. Ela foi também importante para orientação no planejamento e organização da didática utilizada, uma vez que a forma hierárquica dos domínios cognitivos apontou como construir um novo conhecimento e como avaliá-lo. O reflexo deste método foi visto na pequena porcentagem de evasão, na motivação dos alunos em aprender e no alto índice de exercícios realizados.

É válido mencionar que as experiências que a equipe vêm obtendo com o ensino de programação para jovens têm conseguido estabelecer uma base metodológica importante para conduzir ações de ensino eficientes. Neste ponto, as discussões têm procurado apontar melhorias para o formato do curso (duração e número de encontros, por exemplo), a natureza dos estímulos que devem existir entre os encontros presenciais, o uso de redes sociais como meio de aumentar o engajamento, a colaboração e a aprendizagem entre os pares, assim como o uso de outros recursos de apoio às aulas, como é o caso de palestras e visitas a empresas locais de tecnologia.

A ampla aceitação destas ações nas escolas da região é um resultado que permite o amadurecimento de outras ações, principalmente aquelas relacionadas à pesquisa científica. É de grande relevância investigar como o ensino de programação pode ser utilizado para aumentar o interesse dos estudantes por algumas disciplinas ou quais são os impactos gerados para os estudantes quando estes se aproximam de ações articuladas pela universidade. Ao considerar que os estudantes envolvidos no projeto pertencem à rede pública de ensino e que o cenário predominante nas escolas, em geral, é preocupante, dado o desinteresse pelos assuntos escolares e os índices de evasão, percebe-se que os desdobramentos desse projeto podem gerar contribuições mais práticas para a escola, enquanto sistema mais complexo e ecológico.

Este trabalho também trouxe um grande aprendizado para a equipe organizadora, estudantes de Licenciatura em Ciência da Computação da UFPB no campus IV. O curso proporcionou à equipe a vivência de como atuar em um laboratório ensinando programação. Mostrou os desafios no planejamento e na execução das aulas teóricas e das práticas associadas. Além disso, expôs também o benefício de se escolher um instrumento bem definido para guiara prática docente. Por fim, pode-se afirmar que o ensino da linguagem *Scratch* norteada pela taxonomia de Bloom foram fatores de sucesso no ensino de programação de maneira introdutória para alunos do ensino médio.

Referências

- Aureliano, V. C. O.; Tedesco P. C. de A. R. (2012). Avaliando o uso do *Scratch* como abordagem alternativa para o processo de ensino-aprendizagem de programação. In: XXXII Congresso da Sociedade Brasileira de Computação, Curitiba.
- Jesus, E. A. de; Raabe, A. L. A. (2009). Interpretações da Taxonomia de Bloom no Contexto da Programação Introdutória. In: XX Simpósio Brasileiro de Informática na Educação, Florianópolis.
- Krathwohl, D. R. (2002). A revision of bloom's taxonomy: an overview. In: Theory into Practice, n. 41, v. 4, p. 212-218.
- Lifelong Kindergarten (2007). Disponível em: <http://ilk.media.mit.edu/projects.php?id=783>. Acessado em: 23 jul 2013.
- Mélo, F. É. N.; Cunha, R. R. M.; Scolaro, D. R.; Campos, J. L. (2011). Do *Scratch* ao Arduino: Uma Proposta para o Ensino Introdutório de Programação para Cursos Superiores de Tecnologia, In: XXXIX Congresso Brasileiro de Educação em Engenharia, Blumenau.
- Pereira, P. de S.; Medeiros, M.; Menezes, J. W. M. (2012). Análise do *Scratch* como Ferramenta de Auxílio ao Ensino de Programação de Computadores. In: XL Congresso Brasileiro de Educação em Engenharia, Belém.
- Saico, P. D., Lima, A. A., Silva, J. B. B., Azevedo, S., Paiva, L. F., Raposo, E. H., Alencar, Y., Mendes, J. P. (2012). Programação no ensino médio: uma abordagem de ensino orientado ao design com *Scratch*. In: XVII Workshop de Informática na Escola, Rio de Janeiro.
- Scratch (2011). Disponível em: <<http://scratch.mit.edu/about>>. Acessado em: 11 jul 2013.
- Sudol L. A. (2011). Deepening Students Understanding of Algorithms: Effects of Problem Context and Feedback Regarding Algorithmic Abstraction. Carnegie Mellon University.