



VIRTUALIZAÇÃO DO SABERES D'AVÓ: UM MINICURSO DE PROGRAMAÇÃO INTRODUTÓRIA PARA JOVENS

Laíza Ribeiro Silva¹, Dárlinton Barbosa Feres Carvalho¹

¹Universidade Federal de São João del-Rei, laizaribeiroinfor@gmail.com, darlinton@acm.org

Resumo: Diante do progresso das tecnologias, é notável seu impacto causado na vida de todos e a necessidade em saber como lidar com elas. Levar o ensino de programação introdutória para os jovens, ainda em idade escolar, é um importante passo para que sejam inseridos neste mundo de evoluções tecnológicas. Este artigo traz o relato de experiência da virtualização do projeto Saberes D'Avó, que propõe levar o ensino de programação introdutória para os jovens, explorando também o estreitamento de relação entre as gerações. Sua virtualização visa prover uma maior disponibilidade do minicurso a mais jovens, a fim de que eles se tornem produtores de tecnologia, aprendendo assim a moldar seu próprio futuro.

Abstract: On the progress of technology, it is remarkable its impact in everyone's life and the need to know how to deal with them. Offer introductory programming education to adolescents is an important step in order to insert them in the world of technological developments. This article provides an experience-report about the virtualization process of the Saberes D'Avó (Grandma's wisdom) project, which proposes to bring introductory programming education for high school students, also exploring closer relationship between the generations. Its virtualization aims to provide greater availability of the short course to the students, so that they can become producers of technology, thus providing them with new tools in order to shape their own future.

1. Introdução

Diante da evolução e incorporação de tecnologias computacionais na realização de tarefas cotidianas, cada vez mais se torna essencial e necessário a todos dominar técnicas para manipulação e programação de dispositivos digitais. Líderes mundiais, como Vinton Cerf, ex-presidente da ACM e vice-presidente *Chief Internet Evangelist* do Google, e Barack Obama, presidente dos Estados Unidos da América (EUA), enfatizam que ensinar programação para os jovens é um tema de grande relevância e fundamental importância para garantir empregabilidade e inclusão dos mesmos na sociedade no futuro.

Um artigo recente publicado na *Communications of the ACM* (CERF, 2016), Vinton Cerf discute a respeito da importância de saber sobre programação de computadores. Ele enfatiza que este conhecimento faz com que as pessoas tenham uma espécie de mentalidade disciplinada em diversas áreas como engenharia, pesquisas científicas, empresas e política. Cerf diz: “Mesmo se um estudante não seguir carreira em ciência da computação ou área relacionada, estas habilidades



provavelmente serão úteis em qualquer atividade na qual pensamento analítico é útil”¹.

Barack Obama, no discurso *State of the Union* de 2016, apresentou o programa Ciência da Computação para Todos, colocando o ensino de programação em escolas como a grande meta de seu governo para este ano, no intuito de garantir sustentabilidade na empregabilidade de seu povo no futuro. Em entrevista disponível no site Code.org, Obama destaca a importância do ensino de computação para os jovens, ressaltando o quão importante é para os jovens dominarem as ferramentas e tecnologias que mudarão o jeito com que eles farão todas as coisas, enfatizando a importância de eles serem produtores de tecnologia. Em suas palavras: “Não apenas baixe o último app, ajude a projetá-lo. Não apenas use seu smartphone, programe-o. Ninguém nasce um cientista da computação, mas com um pouco de trabalho duro, e alguma matemática e ciência, praticamente qualquer um pode se tornar um. Esta semana é sua chance de tentar. E não deixe qualquer um dizer que ‘você não pode’. Seja você um garoto ou uma garota, se vive na cidade ou em área rural, computadores irão se tornar grande parte do seu futuro. E se você está disposto a trabalhar e estudar com afinco, este futuro é seu para ser moldado”².

Refletindo a respeito destas visões inspiradoras, é perceptível o desafio de articular uma melhor forma para realizá-las na prática, utilizando abordagens que despertem o interesse do público jovem. Também é notável o grande consumo de tecnologia por esse público, bem como o apreço pelo entretenimento digital. Portanto, entendemos ser melhor aliar o ensino de programação a um viés lúdico, especialmente buscando a criação de novos jogos digitais. Dessa forma teremos jovens não apenas consumidores de tecnologias, mas também produtores. Isso motiva-os e mostra-lhes o que são capazes de realizar através da programação, podendo explorar sua criatividade e colocá-la em prática.

Além do viés lúdico, neste trabalho busca-se desenvolver o ensino de programação introdutória também considerando o estreitamento dos laços entre gerações. É enfatizada a importância da contribuição dos mais experientes (idosos), expandindo e enriquecendo o aprendizado com conhecimentos culturais e sociais. Também é almejado capacitar os jovens ainda em idade escolar, para que possam aplicar o aprendizado adquirido a respeito de programação nas tarefas referentes as disciplinas escolares, bem como para interesses pessoais.

Busca-se promover o ensino de programação por meio de uma abordagem interdisciplinar, utilizando o desenvolvimento de aplicativos como ferramentas de intervenção em várias áreas do saber e relacionamentos sociais. Neste sentido, consolida-se a abordagem proposta na forma de um minicurso de programação introdutória, ao qual chamamos de Saberes D’Avó.

Neste sentido, aproveita-se da utilização de plataformas mais modernas para ensino e desenvolvimento de aplicativos móveis, como o *MIT App Inventor*³. O

¹ <http://cacm.acm.org/magazines/2016/3/198866-computer-science-in-the-curriculum/fulltext> - Tradução realizada pelos autores

² <https://www.youtube.com/watch?v=6XvmhE1J9PY> - Tradução realizada pelos autores

³ <http://appinventor.mit.edu/explore/>

desenvolvimento desta ferramenta foi liderado pelo Prof. Hal Abelson (MIT) e Mark Friedman (Google), enquanto Professor Hal estava em sabático na Google em 2009.

Neste artigo é apresentada a criação de um minicurso online da abordagem proposta, ou seja, uma virtualização do Saberes D'Avó. Trabalha-se considerando uma abordagem de ensino que não é dependente da sala de aula para ser executada, e sim do interesse e motivação dos alunos que têm disposição em aprender novas práticas e que entendem o quão positivo são os resultados quando exploradas as habilidades ligadas a programação. É apresentado também um instrumento de avaliação de aprendizado.

O restante deste artigo está organizado como a seguir. A Seção 2 contém a fundamentação teórica considerada neste trabalho. Na Seção 3, é relatado o processo considerado na virtualização do projeto Saberes D'Avó, bem como detalhado a abordagem proposta. Já na Seção 4, está definida a abordagem de ensino utilizada no minicurso online. Por último, na Seção 5, são apresentadas algumas considerações finais sobre o estudo.

2. Fundamentação Teórica

Atualmente, o ensino de programação não faz parte da realidade escolar da grande maioria dos jovens, especialmente os da rede pública, o que se torna um fator preocupante. Com a ascensão do desenvolvimento e uso tecnológico, é cada vez mais importante que os jovens acompanhem esta evolução para estarem adequadamente inseridos na construção da sociedade tecnológica. Segundo Scaico et al. (2014), aprender a programar não deveria ser algo ensinado apenas a estudantes de computação, mas deveria fazer parte do ensino dos alunos ainda em idade escolar. Além de ser algo extremamente relevante por seu fim, vale ressaltar também que o aprendizado a respeito de programação desenvolve habilidades relacionadas a resolução de problemas e a facilidade dos jovens a entender e operar sobre o uso de tecnologias.

Todavia, para que novos recursos possam ser explorados, é necessário que os docentes entendam melhor a importância deles no aprendizado dos alunos. A partir do momento em que os docentes desapegam de metodologias tradicionais e enxergam que além do saber e fazer serem importantes para o aprendizado, a importância do conceito brincar traz uma forma diferenciada de levar conhecimento até aos jovens (THOMAS; BROWN, 2011). Motivar os jovens que iniciam no aprendizado de programação é de suma importância, por desmitificar o assunto, já que programação é tida como algo complexo de se aprender.

Uma das questões de maior dificuldade relatadas pelos iniciantes em programação está na dificuldade em compreender a sintaxe (KELLEHER; PAUSCH, 2005). Logo, para que os jovens se interessem em aprender sobre programação é necessário utilizar linguagens de programação que estejam mais próximas da capacidade de abstração que eles possuem. Para facilitar a compreensão dos iniciantes neste aprendizado, surgiram diversas iniciativas que criaram ambientes de criação de aplicativos baseados no intuito educativo, utilizando linguagens baseadas em blocos. Esses ambientes são temas de pesquisas relevantes, como a realizada



por (CRUZ; LIMA, 2014) em relação aos ambientes Scratch⁴ e Alice⁵.

Podemos observar que, juntamente com o crescimento da importância do uso de tecnologia no dia a dia, a utilização dos smartphones vêm crescendo rapidamente. Portanto, levar até os jovens o ensino de programação voltada para o desenvolvimento de aplicativos para estes smartphones é mais interessante e poderá despertar maior interesse.

O desenvolvimento de aplicativos com intuito educativo já existe há algum tempo (CRUZ; LIMA, 2014). O Instituto Federal do Paraná – Câmpus Irati utilizou o *MIT App Inventor* para o desenvolvimento de atividades de um projeto de extensão. Esse projeto focava principalmente na resolução de problemas matemáticos. Já em nossa pesquisa, alia-se o aprendizado sobre programação introdutória ao estreitamento dos laços entre as gerações, produzindo aplicativos baseados em conhecimentos culturais baseados na experiência de vida dos idosos.

Para que seja possível ensinar programação nas escolas é necessário que haja infraestrutura tecnológica adequada disponível. Ao visitar algumas escolas da região, é notável que isso não é uma realidade. Existem poucos computadores em bom funcionamento e nem sempre a conexão com a internet disponível é de boa qualidade. Diante disso, é necessário considerar também recursos dos próprios alunos, promovendo assim um ensino também fora da escola. Isto é possível já que com o avanço da tecnologia, os computadores e internet encontram-se presentes na grande maioria das residências dos brasileiros. Este trabalho busca aproveitar a tecnologia disponível no cotidiano dos estudantes, como computadores e smartphones como recursos requeridos para a participação no curso.

O ensino na forma online e a distância tem como intuito disseminar o conhecimento para um maior número de pessoas. Os Ambientes Virtuais de Aprendizagem (AVAs) são criados para oferecer um ambiente de ensino para os alunos. AVAs são utilizados na Educação a Distância (EaD), onde os professores montam uma espécie de sala de aula virtual, divulgando materiais para os alunos, bem como tarefas, fóruns de dúvidas, bate-papo, notícias, entre outros.

Podemos afirmar que um ambiente virtual é um espaço fecundo de significação onde seres humanos e objetos técnicos interagem, potencializando, assim, a construção de conhecimentos, logo, a aprendizagem. (DOS SANTOS, 2002, p. 426)

Entendendo o poderoso papel dos AVAs, busca-se utilizá-los para promover o ensino de programação para os alunos como forma de complementar o aprendizado que recebem nas salas de aulas a respeito de outras disciplinas. Espera-se que os alunos aliem o conhecimento adquirido a partir de programação introdutória em outras disciplinas, aplicando-o na prática em suas tarefas dentro da sala de aula e em outros projetos de seus interesses.

Por fim, é necessário definir instrumentos para verificação do aprendizado. Como programação está ligada a conceitos mais abstratos, é ainda mais complexo definir um instrumento que avalie precisamente o grau de aprendizado dos alunos.

⁴ <https://scratch.mit.edu/>

⁵ <http://www.alice.org/>

Bloom (1956) propõe uma taxonomia para definir e escolher instrumentos de avaliação, assim como para definir objetivos instrucionais. Krathwohl (2002) menciona que esta taxonomia, conhecida como taxonomia de Bloom, foi elaborada a fim de facilitar a troca de questões de prova entre professores de diferentes instituições, onde tais questões tivessem o mesmo propósito de avaliação. Já Ferraz e Belhot (2010) complementam dizendo que essa taxonomia também pode ser aplicada para avaliação de qualquer disciplina, principalmente de disciplinas abstratas, como programação introdutória.

Fuller et al. (2007) dizem que apesar de ser a taxonomia mais utilizada, existem categorias difíceis de serem interpretadas. Buscando resolver estas dificuldades, foi definida por Anderson et al. (2001) uma nova versão revisada desta taxonomia, a qual é adotada neste trabalho.

A taxonomia revisada é dividida em duas dimensões: conhecimento e processos cognitivos. Existem verbos associados a cada nível da taxonomia para descrever cada categoria. Eles são utilizados para identificar o nível do aprendizado do aluno a partir da categoria menos complexa até a mais complexa. Esses verbos são: lembrar (reconhecer, relembrar), entender (interpretar, exemplificar, classificar, sumarizar, inferir, comparar, explicar), aplicar (executar, implementar), analisar (diferenciar, organizar, atribuir), avaliar (verificar, criticar) e criar (gerar, planejar, produzir) (KRATHWOHL, 2002).

A Tabela 1 mostra as categorias da taxonomia revisada, seguida por suas interpretações. Vale lembrar que para cada categoria existem diversas interpretações, porém foram consideradas aquelas que se enquadram melhor na avaliação dos alunos que estão tendo contato com o ensino de programação introdutória pela primeira vez.

Tabela 1. Taxonomia de Bloom Revisada e suas interpretações

Categoria	Interpretação da categoria aplicada à programação
Lembrar	Reconhecer a implementação de um determinado conceito.
Entender	Completar partes faltantes de um programa utilizando fragmentos de código.
Aplicar	Resolver um problema familiar, mas com dados ou ferramentas não familiares.
Analisar	Identificar componentes críticos para o desenvolvimento e identificar componentes ou requisitos não importantes.
Avaliar	Encontrar um erro de lógica em um trecho de código dado.
Criar	A partir de blocos de códigos distribuídos aleatoriamente, ordená-los para resolver o problema em questão.

3. Virtualização do Saberes D'Avó

O Saberes D'Avó apresenta uma nova abordagem de ensino de programação introdutória com objetivo de estimular e treinar os jovens a criarem aplicativos Android e, concomitantemente, estreitar os laços entre as gerações. Nessa abordagem, busca-se aproveitar da experiência de vida dos idosos na criação de um aplicativo na forma de um jogo de perguntas e respostas (jogo quiz). Essa abordagem possui um

caráter inovador, adentrando no viés emocional contido na experiência dos idosos e transformando-as de maneira a contribuir para o aprendizado dos jovens a respeito de novas práticas.

Inicialmente, a abordagem foi proposta e realizada na modalidade presencial. Alunos do ensino médio de uma escola participaram de minicurso prático sob orientação de um instrutor. Foram realizadas cinco aulas presenciais em cada minicurso, durante cinco semanas, sendo uma aula de 3h por semana.

A partir desta experiência exitosa, surgiu então o interesse de expandir a aplicação do Saberes D'Avó. Com a intenção de ampliar o alcance, foi proposta uma versão na modalidade semipresencial e logo após na modalidade online. Desta forma, realizou-se uma virtualização da abordagem, ou seja, o desenvolvimento do minicurso online, buscando dar maior escalabilidade na realização do Saberes D'Avó. A Tabela 2 mostra a diferença entre as versões desenvolvidas.

Tabela 2. Detalhamento das três modalidades

Modalidade	Aula	Tutor	Trabalho	Prova
Presencial	Instrutor	Presencial	Em grupo	Presencial
Semipresencial	Tutorial	Presencial/Online	Em grupo	Online
Online	Tutorial	Online	Individual	Online

Para realizar a virtualização do projeto, foi utilizado o Moodle, um AVA para hospedar o material do minicurso e disponibilizar maiores recursos para os alunos. Esse AVA foi escolhido por já ser utilizado pela Universidade Federal de São João del-Rei (UFSJ), em seu Núcleo de Ensino a Distância (NEAD), na Comunidade Acadêmica Virtual (CAV). O NEAD cedeu uma sala virtual na CAV⁶, onde foi possível estruturar o minicurso online.

A fim de dar o maior apoio possível para os participantes, a sala virtual nomeada "Saberes D'Avó – Programação Introdutória", contém dois importantes fóruns. O primeiro chama-se *Fórum de notícias*, utilizado pelos instrutores para informar notícias aos alunos. O segundo fórum foi intitulado *Fórum de dúvidas* e definido como o local apropriado para os alunos sanarem suas dúvidas com os instrutores. Nesse fórum, os participantes podem criar tópicos com suas dúvidas e os instrutores respondem. Essas dúvidas, assim como suas respostas, ficam disponíveis na plataforma Moodle para todos os alunos. Essa ação é importante para ajudar os alunos que também possuem as mesmas dúvidas.

Como meio de ampliar o suporte aos alunos de ambas modalidades, foi criado um grupo secreto no Facebook por se tratar de uma rede social onde a maioria das pessoas têm acesso. Lá são postadas informações sobre as atividades semanais que os alunos têm que cumprir e outras notícias relevantes ao bom desempenho do minicurso. Apesar do Moodle também ter funcionalidades necessárias para divulgação destas informações, a preferência dos alunos por utilizar o Facebook como ferramenta de comunicação reforçou a necessidade de se utilizar também este

⁶ <http://www.campusvirtual.ufsj.edu.br/comunidade/login/index.php>



ambiente para interação entre os participantes.

Para o minicurso, foram desenvolvidos cinco tutoriais referentes a cinco aulas. Para cada um, os alunos têm uma semana para realizar as tarefas, definidas previamente pelos instrutores. Esses tutoriais foram elaborados utilizando o Microsoft Office Power Point 2013 e depois foram exportados para o formato pdf. Cada aula tem em média trinta e quatro lâminas. Escolheu-se utilizar tutoriais em pdf e não vídeoaulas para que o aluno não tenha a necessidade de ficar pausando o vídeo enquanto realiza as tarefas. Como programação é um assunto novo para os participantes, é importante que eles avancem quando estiverem seguros a prosseguir com o conteúdo disponibilizado.

A próxima seção apresenta detalhes sobre os conteúdos desenvolvidos para o minicurso, bem como é proposta a abordagem de ensino como um todo. Vale lembrar que apesar desta proposta ser baseada na experiência do minicurso presencial, são apresentadas adaptações e novidades, conforme a seguir.

4. Abordagem de Ensino

Primeiramente, vale ressaltar que a abordagem de ensino utilizada nas modalidades semipresencial e online é baseada fundamentalmente nos mesmos princípios utilizados na modalidade presencial. Embora uma diferença importante é que para a modalidade presencial os tutoriais não foram utilizados. Na modalidade presencial os instrutores explicavam as tarefas semanais durante as aulas, e os alunos desenvolviam as atividades propostas na própria aula. A partir das modalidades semipresencial e online, consolidou-se esta abordagem em tutoriais disponíveis na versão pdf na plataforma CAV, no AVA Moodle.

Foram elaborados cinco tutoriais para o minicurso, além de um tutorial extra ensinando quais procedimentos são necessários para que baixar e testar o aplicativo desenvolvido. Como já havia sido realizado aplicação do minicurso na modalidade presencial, não foi difícil construir os tutoriais com os conteúdos para as aulas. São exploradas questões de disciplinas de matemática, ciência, história e cultura regional para ilustrar exemplos utilizados no ensino dos conceitos básicos de programação. O trabalho final proposto é a criação de um jogo do tipo quiz (perguntas e respostas), considerando o conteúdo obtido a partir de entrevistas com idosos, especificamente, seus avós.

Considerando que programação de computadores não era conhecida previamente pelos alunos, e a fim de motivá-los e mostrá-los que são capazes de desenvolver com êxito as atividades propostas, foi definido que na primeira semana do minicurso eles desenvolveriam uma calculadora simples. Por ser uma tarefa trivial, até mesmo para quem nunca teve contato com programação, propor o desenvolvimento de uma calculadora faz com que os alunos acreditem que são capazes de fazer as atividades propostas, porém agora com a motivação e a certeza que programar não é algo tão complexo quanto imaginavam.

O fluxograma apresentado na Figura 1 mostra as etapas realizadas na aplicação da abordagem de ensino proposta.

Por fim, acreditando na importância da motivação para que os alunos

desenvolvam bons trabalhos, eles são convidados a participar de um desafio, intitulado “Tarefa Premiada”. Neste desafio, o aplicativo mais bem trabalhado é publicado no Google Play Store, obedecendo os critérios de funcionalidade e inovação. Desta forma, premia-se os melhores trabalhos dos melhores alunos.

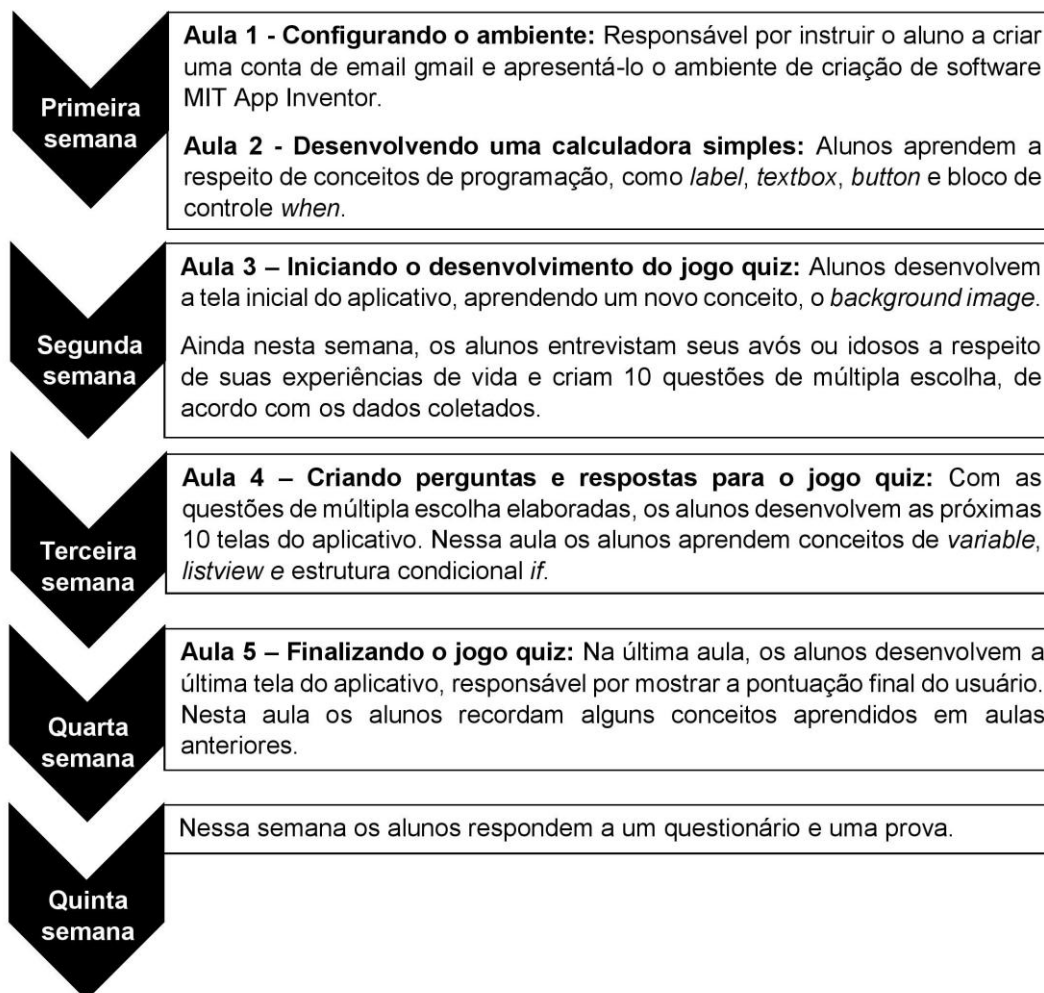


Figura 1. Etapas da abordagem de ensino proposta

5. Considerações Finais

Neste artigo é apresentada uma evolução da abordagem inovadora do Saberes D’Avó, proposta para levar o ensino de programação introdutória aos jovens a partir do estreitamento da relação entre as gerações. De modo a aumentar o alcance da abordagem proposta, apresentamos o processo de sua virtualização, o qual consistiu na criação de um minicurso semipresencial e online.

A modalidade presencial, responsável pelo surgimento das outras modalidades, contou com quatorze participantes, a semipresencial com oito e à distância com dez. Após a conclusão do minicurso nas três modalidades, os alunos responderam uma prova com vinte e quatro questões. Essas questões foram formuladas de acordo com a taxonomia de Bloom revisada, para que assim fosse possível mensurar o nível de aprendizado dos alunos após terem concluído o



minicurso. Para isso, foram elaboradas quatro perguntas para cada uma das seis categorias, totalizando assim uma prova de vinte e quatro questões.

O gráfico da Figura 2 mostra a porcentagem de acertos em cada categoria da taxonomia de Bloom revisada. É notável que a modalidade semipresencial superou a porcentagem da média de acertos de todas as categorias da taxonomia, seguida da modalidade online e por último da modalidade presencial.

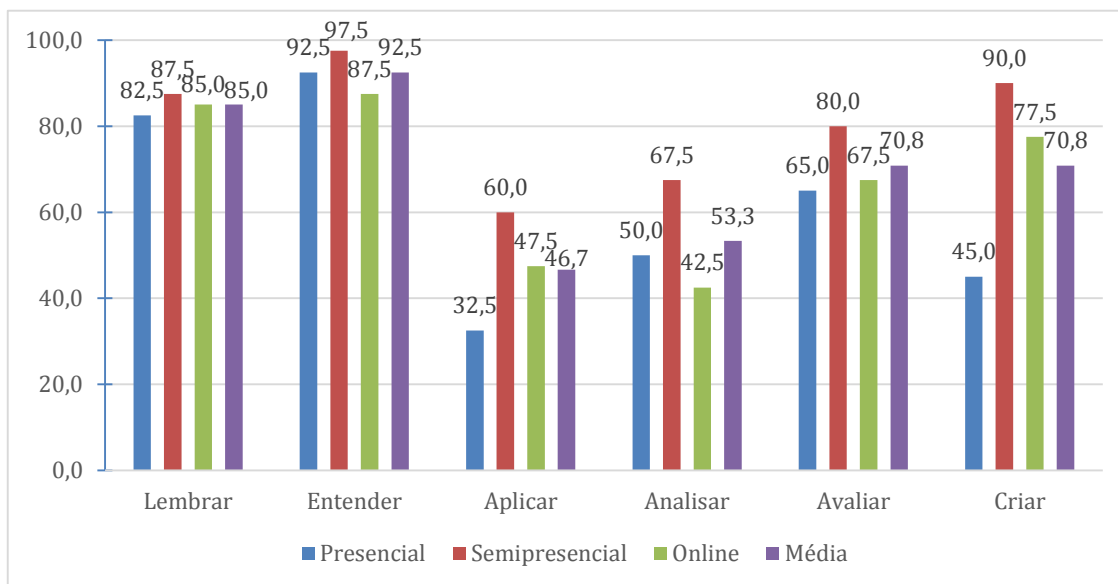


Figura 2. Comparação da porcentagem da média de acertos entre todas as modalidades

A aplicação semipresencial teve suas particularidades. Ela pode ser vista como uma junção de parte da aplicação presencial e online. Acredita-se que o diferencial dessa modalidade tenha sido a disponibilização de tutoriais, com a explicação detalhada das tarefas a serem realizadas nas aulas. Concomitantemente, a presença de um instrutor nos encontros semanais para tirar a dúvida dos alunos e o trabalho em grupo são fatores importantes no impacto do aprendizado. A abordagem semipresencial dá uma liberdade de aprendizado ao aluno, pois ele realiza as tarefas no seu tempo e de acordo com as suas limitações. Logo, se ele não tem facilidade em aprender determinado conteúdo, ele tem a oportunidade de revisar quantas vezes forem necessárias antes de concluir a atividade. O relacionamento entre os membros do grupo também é muito importante na aprendizagem, já que um pode ajudar o outro a superar suas dificuldades.

Vale ressaltar que os alunos tiveram um total de 2 horas para realizar a avaliação, tendo 5 minutos para resolver cada questão. Ao final do minicurso, os alunos responderam a um questionário online onde foram elaboradas 11 perguntas com o objetivo de inspecionar o nível de conhecimento dos alunos antes e após participarem do minicurso, além de opiniões dos mesmos sobre a inserção da programação no dia a dia escolar. De acordo com o questionário, os alunos de todas as modalidades, em sua maioria, declararam ter conhecimento básico da língua inglesa (62,8%), não sabiam programar (78,5%) e também não conheciam o *MIT App Inventor* (91%). Todos os alunos gostariam da inclusão de uma disciplina de programação na grade curricular do MEC e declararam acreditar que aprender a

programar melhora a criatividade e o interesse dos mesmos em relação a outras disciplinas obrigatórias. Vale ainda destacar que os alunos mostraram interesse em utilizar o aprendizado de programação para resolver problemas de outras disciplinas, dando a este ensino um sentido interdisciplinar.

O minicurso apresentado neste artigo é uma possibilidade de inserir o ensino de programação paralelamente com o ensino fundamental e médio nas escolas brasileiras. Espera-se que além de aprender sobre programação introdutória, aproveitem em sala de aula aplicando o conhecimento para realizar trabalhos e resolver tarefas das disciplinas nos quais estão matriculados.

Referências Bibliográficas

- ANDERSON, L. W.; KRATHWOHL, D. R.; BLOOM, B. S. *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. [S.l.]: Allyn & Bacon, 2001.
- CERF, V. G. Computer Science in the Curriculum. *Communications of the ACM*, 59(3):7–7, 2016.
- DA CRUZ, ALLAN KÁSSIO BECKMAN SOARES; LIMA, LEANDRO CAVALCANTE MENDONÇA. Estudo e testes de usabilidade em sistemas de autoria de software: scratch e alice. *Blucher Design Proceedings*, v. 1, n. 4, p. 3673-3685, 2014.
- DOS SANTOS, EDMÉA OLIVEIRA. "Ambientes virtuais de aprendizagem: por autorias livres, plurais e gratuitas." *Educação e Contemporaneidade* 11.18: 425-435, 2002.
- FERRAZ, A. P. C. M. et al. Taxonomia de Bloom: revisão teórica e apresentação das adequações do instrumento para definição de objetivos instrucionais. *Gest. Prod., São Carlos*, v. 17, n. 2, p. 421-431, 2010.
- FULLER, U.; JOHNSON, C. G.; AHONIEMI, T.; CUKIERMAN, D.; HERNÁNLOSADA, I.; JACKOVA, J.; LAHTINEN, E.; LEWIS, T. L.; THOMPSON, D. M.; RIEDESEL, C. et al. Developing a computer science-specific learning taxonomy. In: *ACM. ACM SIGCSE Bulletin*. [S.l.], v. 39, n. 4, p. 152–170, 2007.
- KRATHWOHL, D. R. A revision of bloom's taxonomy: An overview. *Theory into practice*, Taylor & Francis, v. 41, n. 4, p. 212–218, 2002.
- KELLEHER, C. AND PAUSCH, R. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR)*, 37(2):83–137, 2005.
- SCAICO, P. D., DE LIMA, A. A., AZEVEDO, S., DA SILVA, J. B. B., RAPOSO, E. H., ALENCAR, Y., MENDES, J. P., SCAICO, A., ET AL. Ensino de programação no ensino médio: Uma abordagem orientada ao design com a linguagem scratch. *Revista Brasileira de Informática na Educação*, 21(02):92, 2013.
- THOMAS, D. AND BROWN, J. S. *A new culture of learning: Cultivating the imagination for a world of constant change*, v. 219. CreateSpace Lexington, KY, 2011.