

Percepções de Professores sobre Recursos de Ajuda em Ferramentas de Programação baseadas em Blocos

Carmen Vera Scorsatto Brezolin^{1,2}, Milene Selbach Silveira¹

¹Escola Politécnica — Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Avenida Ipiranga 6681 – Porto Alegre – RS – Brasil

²Instituto Federal Sul-Rio-Grandense (IFSUL) – Passo Fundo – RS – Brasil

carmen.brezolin@edu.pucrs.br, milene.silveira@pucrs.br

Abstract. *There are currently many tools with a proposal to assist in the development of programming skills and computational thinking. However, there are also interaction difficulties regarding their usage. Hence, the help systems within these tools become crucial when we consider their usefulness and their variety of users. In this scenario, we investigate the users' perception of this type of resource. To accomplish this, we adopted an exploratory methodology. We sent an online questionnaire and a total of 95 participants attended our research. Among the results obtained, we highlight the most used tools, the resources the participants find most helpful and productive, difficulties encountered, other ways to seek help, and suggestions for new resources.*

Resumo. *Existem, nos dias atuais, muitas ferramentas com a proposta de auxiliar no desenvolvimento de habilidades de programação e do pensamento computacional. Porém, observam-se dificuldades na interação com essas ferramentas e, considerando a importância de uso e os diversos públicos que elas atendem, os recursos de ajuda nelas disponíveis se tornam muito importantes. Nesse contexto, o objetivo deste trabalho é investigar a percepção de seus usuários - neste caso, os professores - sobre este tipo de recurso. A metodologia utilizada foi de caráter exploratório, sendo os dados coletados por meio de um questionário on-line, o qual foi respondido por 95 participantes. Dentre os resultados obtidos, destacam-se as ferramentas mais utilizadas, os recursos que acham mais úteis e produtivos, além de dificuldades encontradas, outras formas para buscar por ajuda e sugestões de novos recursos.*

1. Introdução

Nos dias atuais, existem muitas ferramentas que se propõem a desenvolver habilidades de programação e o pensamento computacional, assim como surgem muitas propostas pedagógicas que fazem uso de ferramentas de programação baseadas em blocos para contribuir nos processos de ensino e de aprendizagem [Neto 2013, Charao e Ritter 2020].

Apesar de estas ferramentas serem recursos educacionais que têm se mostrado promissores, observam-se dificuldades em seu uso [Neto 2013, Ribeiro 2019, Charao e Ritter 2020, Assunção et al. 2021] e necessidade de apoio adicional por parte dos alunos [Wiggins et al. 2021]. Dentre as dificuldades, encontra-se a necessidade da junção de muitos blocos de comando (dependendo do que se deseja fazer), de saber qual

comando usar para criar algo e de se utilizar alguns comandos como os associados à estrutura de decisão e repetição e os operadores, por exemplo [Neto 2013].

Assim, considerando a importância de uso deste tipo de ferramenta e os diversos públicos que elas atendem, os recursos de ajuda se tornam muito importantes. Segundo Patrick e McGurgan (1993), os recursos de ajuda podem preencher a lacuna entre a complexidade do software e a necessidade de simplicidade do usuário. Mas, apesar do potencial que estes recursos têm em apoiar os usuários a se apropriarem e, de melhor forma, utilizarem as aplicações que necessitam, para Welty (2011), as pessoas não estão satisfeitas com a ajuda disponível. Segundo Vouligny e Robert (2005), esses sistemas geralmente fornecem informações muito genéricas e o usuário necessita ter conhecimento suficiente para adaptar as informações a sua situação particular, acabando, assim, mais satisfeitos com a ajuda fornecida pela Internet do que com a própria ajuda do software. Ainda, de acordo com Novick et al. (2009), os sistemas de ajuda não são significativamente mais úteis do que resolver problemas por “Tentativa e Erro” (T&E).

É nesse contexto, sobre o uso de recursos de ajuda em ferramentas de programação baseadas em blocos, que esse trabalho se insere. O objetivo é investigar a percepção de seus usuários sobre as ajudas disponíveis neste tipo de ferramenta e seus impactos no aprendizado tanto das ferramentas em si quanto dos conceitos relacionados a algoritmos e programação.

Para apresentar os resultados desta pesquisa, este trabalho está organizado da seguinte forma: na Seção 2 é apresentado o referencial teórico; na Seção 3, a análise dos recursos de ajuda; na Seção 4, uma discussão sobre os resultados obtidos; na Seção 5, dificuldades e perspectivas futuras e, por fim, na Seção 6, as considerações finais desse trabalho, seguidas pelas referências utilizadas em sua construção.

2. Referencial Teórico

Nessa Seção, serão apresentados conceitos e estudos relacionados a sistemas de ajuda e a ferramentas de programação baseadas em blocos.

2.1. Sistemas de Ajuda

No final dos anos 80, Mckita (1988) previu que, com a chegada de sistemas de computador com interfaces de usuários orientada a objetos e com inteligência artificial, estes seriam tão intuitivos e tão fáceis de aprender que os usuários não precisariam de treinamento e nem de documentação para usá-los. Porém, para Sales e Queiroz Palmeira (2019), é improvável que uma interface possa se comunicar por si só e seja totalmente intuitiva, para qualquer tarefa e usuário.

Neste contexto, apesar dos recursos existentes, segundo Welty (2011), muitos usuários acabam, quase que exclusivamente, usando motores de busca para obter ajuda no lugar dos recursos oferecidos pelas aplicações. Essa preferência, segundo o autor, se deve à excelência das suas respostas às consultas, por encontrarem os sites relevantes, fóruns e *blogs* que os sistemas de ajuda geralmente não encontrariam, além de apresentarem uma interface única, muitas vezes familiar ao usuário.

Por fim, Chilana et al. (2013) acreditam na necessidade de se conhecer melhor o comportamento de busca de ajuda pelos usuários, em diferentes contextos, além de se explorar estratégias para fornecer ajuda mais relevante no contexto de suas tarefas.

2.2. Ferramentas de Programação em Blocos

Programação em blocos se caracteriza por utilizar blocos visuais que podem ser arrastados para a montagem dos programas, funcionando a partir da ação de arrastar e soltar esses blocos num formato de quebra-cabeça, em que as peças se encaixam (Figura 1). Essas ferramentas não requerem domínio da sintaxe, como é necessário em linguagens textuais como C++, Javascript, Java e Python.

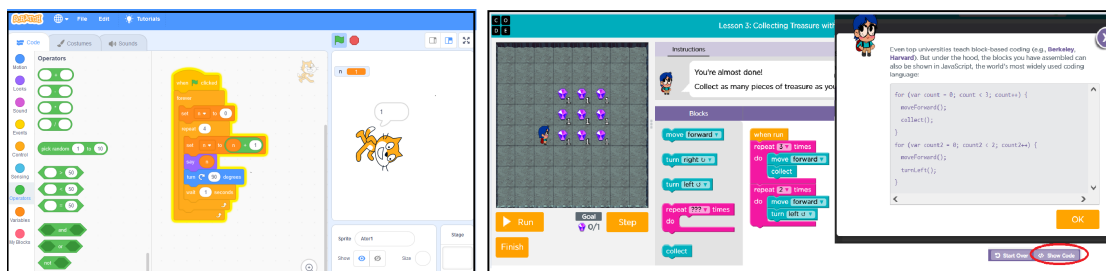


Figura 1. Encaixes característicos dos blocos do Scratch e Code.org

De acordo com Souza et al. (2021), é importante utilizar ferramentas que possam dinamizar o processo de aprendizagem criativa, ao ensinar programação para crianças. E, segundo uma análise dos autores, as ferramentas educacionais mais utilizadas para propor o ensino de programação para crianças foram: Scratch, Arduino, Code.org e o jogo Lightbot, o que é confirmado no trabalho de Brezolin e Silveira (2021), no qual encontraram três dessas quatro ferramentas (Scratch, Code.org e o jogo Lightbot) como as mais citadas em um mapeamento sistemático realizado.

3. Análise de Recursos de Ajuda: a pesquisa e seus participantes

A fim de investigar a percepção de professores sobre as ajudas disponíveis neste tipo de ferramentas e seus impactos no aprendizado tanto das ferramentas em si quanto dos conceitos relacionados a algoritmos e programação, foi realizada uma pesquisa por meio de um questionário on-line. Destaca-se que esta pesquisa foi submetida e aprovada pelo Comitê de Ética em Pesquisa da Universidade².

Este questionário foi aplicado a professores que trabalham ou já trabalharam com ensino de programação/pensamento computacional e que utilizam este tipo de ferramentas, sendo divulgado pela web, por e-mail, por grupos de WhatsApp e por redes sociais.

O questionário on-line aplicado foi elaborado com questões divididas em 6 seções:

- 1- **Perfil do Participante**, foco na identificação do participante;
- 2- **Atuação em sala de aula**, foco no tempo de atuação em sala de aula e no nível de maior atuação;
- 3- **Área de atuação**, foco na área de atuação, em relação ao ensino de Lógica de Programação e/ou Pensamento Computacional;
- 4- **Uso de Ferramentas para apoio**, foco nas ferramentas utilizadas para ensino de Lógica de Programação e/ou Pensamento Computacional;
- 5- **Uso de recursos de ajuda**, foco nos recursos de ajuda (já) utilizados;

²Projeto cadastrado com N° CAAE: 54357721.0.0000.5336

6- Possibilidades futuras dos recursos de ajuda, foco em sugestões e observações sobre o tema.

Todas as questões concedem, ao participante, a alternativa de não responder, ou seja, não são obrigatórias.

Responderam ao questionário 95 participantes, sendo 48 (50,5%) do sexo masculino, 46 (48,4%) do sexo feminino e 1 (1,1%) que preferiu não informar. Pode-se observar, conforme a Figura 2 (a), que o maior número de participantes se encontra na faixa entre 40 e 50 anos, seguida pela faixa entre 30 e 40. Sobre a titulação, conforme pode ser visto na Figura 2 (b), pode-se observar que Graduação (29,5%), Especialização (28,4%) e Mestrado (26,3%) estão distribuídos quase de maneira uniforme e representam a maior parte da formação de maior nível dos participantes (84,2%).

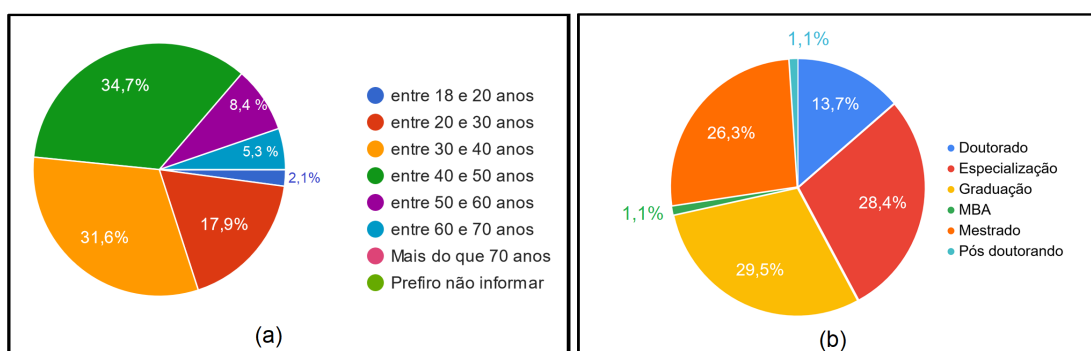


Figura 2. Faixa etária e titulação dos participantes

Sobre a questão de o participante **atuar em sala de aula**, 81,1% dos participantes afirmaram estar atuando ou já ter atuado. Conforme ilustrado na Figura 3 (a), a maior parte dos participantes possui mais de 10 anos de experiência em sala de aula (68%). Se destaca, também, o número de professores que atuam no ensino médio (28,6%) e ensino superior (27,3%), totalizando 55,9%.

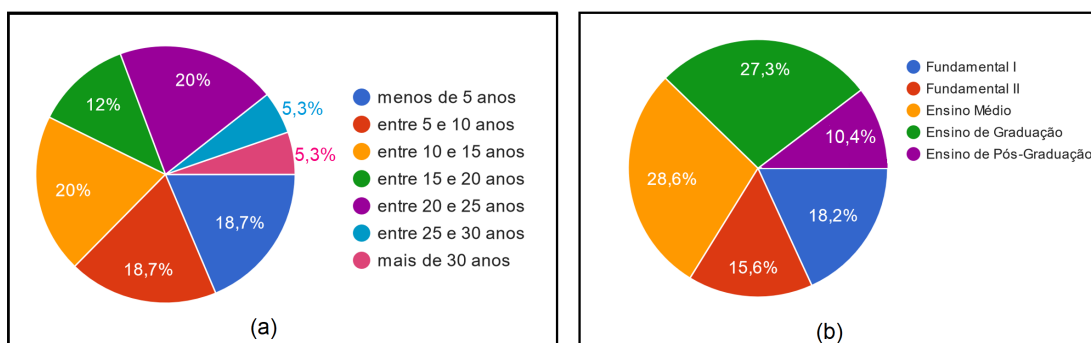


Figura 3. Tempo e nível de atuação em sala de aula

Referente aos **tópicos abordados**, 68,4% informaram que abordaram os tópicos de lógica de programação ou pensamento computacional. Já sobre as **ferramentas utilizadas**, as seis mais citadas foram o Scratch, o Code.org, o Arduino, o App Inventor, Blockly e o Alice (Tabela 1). Além delas, ainda foram mencionadas, com 1,6% cada: Portugol Studio - Construct 3 - Cubetto - mBlock - Kodular - Tinkercad - Code Combat - RenPy - Turtle - Ardublock - WeDo 2.0 - Mindstorm Ev3 - Kodu - Greenfoot - Stencyl .

Tabela 1. Ferramentas utilizadas

Descrição	Percentual
Scratch	81,3%
Code.org	50,0%
Arduino	43,8%
App Inventor	37,5%
Blockly	15,6%
Alice	6,3%

4. Sobre os Recursos de Ajuda: resultados da pesquisa

Quanto ao **uso dos recursos de ajuda** da própria ferramenta, 72,3% afirmaram que já utilizaram e 27,7% que não.

Quanto ao **tipo de recurso de ajuda** que os participantes consideram mais útil, a preferência é por vídeos (74,2%); em segundo, tutoriais das ferramentas (69,7%); em terceiro, os exemplos prontos (66,7%); em quarto, sites de perguntas e respostas/fóruns de discussões (53%); em quinto, dicas durante o uso da ferramenta (45,5%); em sexto, pedir ajuda de um colega (34,8%); em sétimo, FAQ (Perguntas Respondidas Frequentemente) (28,8%); e, em oitavo, o uso de redes sociais (9,1%). Outras opções, como e-mail, contato com suporte ao usuário, documentação e pesquisar no Google, foram poucas vezes mencionadas como um recurso útil.

Os participantes, que afirmaram utilizar os recursos de ajuda da própria ferramenta, apresentaram as **formas como lhes auxiliaram** (Tabela 2).

Tabela 2. Forma pela qual os recursos auxiliaram a sanar suas dúvidas

Descrição	Percentual
tutoriais	36%
exemplos de código semelhantes	26%
resumo das funcionalidades da ferramenta	10%
explicações detalhadas da sintaxe e comandos	7%
dúvidas de outros usuários nos fóruns	5%
ideias de Objetos de Aprendizagem.	2%

Dos 42 participantes que responderam a questão referente à forma que utilizaram os recursos de ajuda da própria ferramenta para sanar suas dúvidas, 7 participantes mencionaram os motivos pela preferência por recursos de ajuda que não são da ferramenta:

- “Muitas vezes vídeos tutoriais de usuários mais experientes são as maiores fontes de ajuda”;
- “Em outras ferramentas como o Open Roberta não encontrei muita informação na plataforma e preferi assistir tutoriais externos de usuários no Youtube”;
- “Na maioria das vezes busquei outros tutoriais na web”;
- “A ajuda do Arduino não existe, dependendo da IDE o que tem sim é bastante documentação e exemplos (quase tudo em inglês)”;
- “Lembro que ajudaram, mas sempre tinha que procurar bastante até encontrar”;
- “Prefiro Vídeos no YouTube”;
- “Algumas documentações ainda são bastante escassas”.

Referente à **forma de apresentação de ajuda**, a preferência é por vídeos (69,7%), seguida da ajuda por texto, com 13,6%, e imagens, com 10,6%. Outros formatos como exemplos de projetos, histórias em quadrinhos e vídeos curtos apareceram de maneira discreta, representando 1,5% cada (Figura 4 (a)).

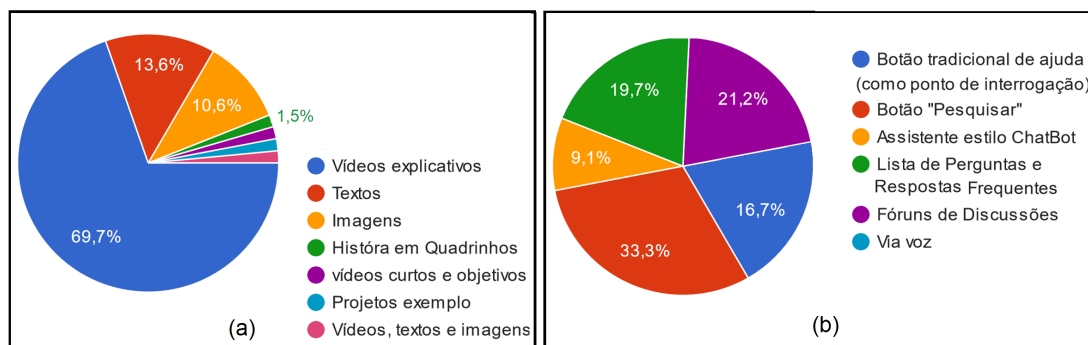


Figura 4. Forma de apresentação e acesso ao conteúdo

A **forma de acessar** aos recursos de ajuda considerada mais produtiva pelos participantes foi o botão “Pesquisar”, com 33,3%, seguida pelos fóruns de discussões, lista de Perguntas e Respostas Frequentes, o botão tradicional de Ajuda (como o ponto de interrogação) e assistentes como os *chatbots* (Figura 4 (b)).

A opinião dos participantes sobre as **dicas** que algumas ferramentas oferecem durante o uso ficaram divididas: dos 52 que responderam a essa questão, 48% afirmaram gostar das dicas disponibilizadas e 52% afirmaram não gostar. Os participantes apresentaram diversos argumentos para justificar suas escolhas, estando os principais pontos listados na Tabela 3.

Tabela 3. Opinião sobre as dicas que algumas ferramentas oferecem

Sim gostam (48%)	Não gostam (52%)
“Auxilia na compreensão e uso da ferramenta”	“Apresentadas de forma genérica e não específica”
“Interessante para usuários em primeiro acesso”	“Chato por ser invasivo (pop-ups sem eu pedir)”
“Dão alguma ajuda contextual”	“Geralmente não parece útil”
“Deixa claro e facilita o uso”	“Tira o foco do estudante”
“Apresentam dicas de prevenção de erros”	“Complexas e pouco didáticas”
“Informações sobre boas práticas de uso”	“Texto e em inglês”
“São bem intuitivas e claras”	“Precisa usar vocabulário menos técnico”
“Auxiliam o usuário da ferramenta em sua utilização”	“Tentam adivinhar o que você está tentando fazer e via de regra erram”

Sobre a questão referente à **troca de experiências** de uso destas ferramentas, 62 participantes deram sua opinião. Desses, 68% responderam que sim. Quanto a com quem costumam trocar experiências, 72% dos respondentes afirmaram ser com colegas de trabalho ou faculdade, 14% afirmaram ser com os próprios alunos, 10% mencionaram fóruns e redes sociais e 4% citaram trocar experiências durante os cursos.

Referente à forma em que os participantes costumam fazer essas trocas, as mais citadas foram: “*Conversa presencial - Rodas de conversa - Conversas de corredor*”, “*Por e-mail - WhatsApp*”, “*Reuniões via google Meet*”, “*Fóruns ou grupos em redes sociais*”, “*Cursos de formação docente*” e “*Grupos de trabalho e de Pesquisa*” .

Sobre os **tipos** de recurso de ajuda **idealizados** para ferramentas de apoio ao ensino de lógica de programação ou pensamento computacional, as sugestões estão categorizadas e listadas na Tabela 4. Ficou evidente a preferência por vídeos, exemplos de uso, pequenos tutoriais e recursos no formato de Perguntas e Respostas.

Tabela 4. Tipo de recurso de ajuda idealizados

Descrição	Nº recorrência
Vídeos explicativos	10
Exemplos de uso	8
Perguntas e respostas	5
Pequenos tutoriais/Tutoriais de imagem e vídeo	4
Ajuda de acordo com o contexto	3
Verificação de erros ou explicação da não realização da ação	2
Tutoriais gamificados	2
Chatbot	2
Recurso de atendimento remoto	2
Ferramentas intuitivas	2
Unir foruns com a ferramenta	1
Cursos e minicursos	1
Interação entre pessoas	1
Algo que consiga entender a real dificuldade e carência do usuário	1
Algo que sugerisse melhoria no código	1
Documentação colaborativa	1

Dos participantes que responderam a questão sobre os recursos idealizados, um participante descreveu que os recursos de ajuda atuais são suficientes e, outro, que nunca foi uma preocupação no desenvolvimento das aulas.

Referente a **sugestões e observações nesse contexto** (de apoio ao ensino de lógica de programação ou pensamento computacional), destaca-se o apresentado na Tabela 5.

Tabela 5. Sugestões e Observações

Descrição	Nº recorrência
Estimular o engajamento e troca experiências entre alunos e professores	3
Perguntas e respostas mais objetivas/FAQ bem feito	2
Vídeos explicativos e didáticos com exemplos claros	1
Melhor suporte para idiomas como o português do Brasil	1
“Ajuda” mais visual, não apenas com textos	1
Tutoriais de instalação/configuração e uso de funcionalidades	1
Cursos para professores	1
Mais preocupação com as reais necessidades dos usuários	1
Mais acessibilidade dos recursos das ferramentas	1
Utilizar essas ferramentas para planejar as tarefas (exemplo matemáticas)	1

Em termos mais gerais, não específico a uma ferramenta em si, um participante mencionou a questão referente a importância de respeitar o tempo de aprendizagem: *“Acho que é importante a pessoa que está aprendendo saber que será perfeitamente normal encontrar dificuldades e situações em que ela poderá “travar” enquanto aprende Lógica de Programação. Botar a mão na massa e entender um pouquinho de cada vez é essencial para passar por esses momentos difíceis”*.

5. Dificuldades e Perspectivas futuras: discussão

Ao avaliar os resultados do questionário on-line aplicado, observa-se que algumas questões se sobressaíram nessa análise, as quais serão discutidas nessa Seção.

Uma das questões que surgiu foi quanto ao tipo de recursos de ajuda que os participantes consideram mais útil. Houve um baixo número de participantes que selecionaram a opção de Pesquisar no Google (1,5%), o que, segundo Welty (2011), seria uma preferência dos usuários. Talvez o foco diferente interfira nas preferências: aqui participantes professores e 70,6% deles com algum tipo de pós-graduação e, no estudo de Welty, estudantes de graduação e ajuda de softwares com perfil de uso comercial (.NET e Java).

Já os participantes que afirmaram utilizar os recursos de ajuda da própria ferramenta (72,3%) salientaram a importância dos tutoriais disponibilizados e que, por meio deles, *“foi possível dar os primeiros passos na ferramenta”* e *“entender as funcionalidades da ferramenta”*, além da importância dos exemplos disponibilizados nesses recursos.

Nesse sentido, dado o número de participantes que afirmaram utilizar os recursos de ajuda das próprias ferramentas, compatibilizamos com a afirmação de Sales e Queiroz Palmeras (2019). Estes autores citam ser improvável que uma interface possa se comunicar por si só e seja totalmente intuitiva, contrariando as previsões de Mckita (1988), quando diz que com a evolução dos sistemas, a documentação e treinamento seriam desnecessários. Os dados apresentados mostram que, além de 72,3% dos participantes admitirem ter usado os recursos de ajuda, nas sugestões/observações eles mencionam cursos, mostrando que ainda precisamos de *“treinamento”* e de documentação de ajuda, para usar os sistemas atuais.

Se destacaram, como motivos pela preferência por recursos de ajuda que não são da própria ferramenta, a falta de disponibilidade de recursos em algumas delas, a vantagem de materiais disponibilizados por usuários experientes, as dificuldades com o idioma (*“quase tudo em inglês”*) e a dificuldade de encontrar e de manusear esses recursos. Esses motivos, destacados pelos participantes, estão de acordo com as afirmações de Nielsen (1994), quando diz que as informações devem ser fáceis de pesquisar, devem ser focadas na tarefa do usuário, devem listar as etapas concretas a serem realizadas e não devem ser muito extensas.

A opinião dos participantes sobre as dicas que algumas ferramentas oferecem ficou dividida, 48% afirmaram gostar das dicas disponibilizadas e 52% afirmaram não gostar, o que de certa forma está de acordo com o estudo de Marwan et al. (2020), no qual os participantes preferem solicitar ajuda sob demanda, já que lhes dá controle sobre quando receber dicas. Assim, a presença dessas dicas precisa ser configurável, dando o poder para o usuário escolher se prefere receber essas dicas ou não.

Sobre a questão referente a troca de experiências de uso destas ferramentas, 72% dos participantes afirmam que estas ocorrem com colegas de trabalho ou faculdade, o que reforça a importância da interação entre as pessoas. Nesse contexto, de acordo com estudo de Price et al. (2017), ao investigar o processo de quando a ajuda é fornecida por um ser humano ou por um computador, os autores concluem que os participantes consideram a ajuda de um tutor humano mais confiável, perceptiva e interpretável.

A partir destes resultados, surgiram argumentos sobre as necessidades dos

participantes, os quais são sumarizados na Tabela 6.

Tabela 6. Apontamentos referentes aos recursos de ajuda

Item	Descrição
Objetividade dos conteúdos	Fundamental a objetividade dos conteúdos apresentados nos recursos de ajuda
Tutoriais pequenos	Pequenos tutoriais (tamanho menor dos materiais) de ajuda parecem facilitar a usabilidade desses
Formato Visual	Preferência por um formato mais “visual” com o uso de vídeos
Exemplos	Destaque para exemplos de uso dos comandos
Interação entre as pessoa	Promoção de interação e engajamento entre as pessoas
Gamificação	Tutoriais Gamificados podem incentivar ao uso de ajuda
Simular a Comunicação humana	Recursos de ajuda que simulem comunicação humana “efeito persona” ¹ podem apoiar a preferência por trocas de experiências com colegas

6. Considerações Finais

Considerando a importância de uso das ferramentas de programação baseadas em blocos no ensino de lógica de programação e pensamento computacional e, ainda, os mais diversos níveis de educação em que estas são usadas, os recursos de ajuda se tornam muito importantes. Nesse contexto, este trabalho teve como objetivo investigar a percepção dos usuários - neste caso, os professores - sobre os recursos de ajuda disponibilizados nas Ferramentas de Programação baseadas em Blocos.

Os dados foram coletados por meio de um questionário on-line, o qual foi respondido por 95 participantes. Dentre os resultados obtidos, destacam-se as ferramentas mais utilizadas, o formato de recursos de ajuda que consideram mais útil e o tipo de recurso de ajuda que idealizam para essas ferramentas, além de sugestões/observações nesse contexto. Com base nesses resultados, destacamos alguns apontamentos referentes a perspectivas futuras dos recursos de ajuda. Apontamentos esses que visam promover a reflexão sobre a construção de recursos de ajuda neste contexto, seus impactos no aprendizado, incentivo de uso, tanto das ferramentas em si quanto dos conceitos relacionados a algoritmos e programação.

Uma das limitações deste trabalho se trata do formato de questionário. Pelo fato de ter sido on-line e sem acompanhamento de um entrevistador, muitas das questões não foram respondidas e, assim, fica difícil garantir uma cobertura adequada do uso dos recursos de ajuda das ferramentas. Outra limitação é sobre a amplitude do tema; existem várias formas de entendimento sobre o que são recursos de ajuda e como esses recursos podem ser implementados nas ferramentas de forma efetiva. Como continuidade aos estudos sobre o tema, pretendemos avaliar o uso destes recursos por meio de observação de seus usuários - professores e estudantes - com a realização de oficinas.

Agradecimentos

Este estudo foi parcialmente financiado pela Uol EdTech.

¹efeito persona : quando a presença de um personagem animado em um ambiente interativo, mesmo não expressivo, parece influenciar a percepção dos sujeitos [Lester et al. 1997].

Referências

- Assunção, O. B., Prates, R. O., e França, E. S. (2021). Relato da aplicação de uma sequência didática fundamentada nas metáforas de perspectivas culturais para fomentação do pensamento computacional. In *Anais do XXIX Workshop sobre Educação em Computação*, pages 131–140. SBC.
- Brezolin, C. S. e Silveira, M. (2021). Panorama brasileiro de uso de ferramentas para desenvolvimento do pensamento computacional e ensino de programação. In *Anais do XXIX Workshop sobre Educação em Computação*, pages 398–407, Porto Alegre, RS, Brasil. SBC.
- Charao, A. S. e Ritter, F. (2020). Investigando dificuldades em recursos do code. org: Aplicação do método do percurso cognitivo ao tutorial artista da hora do código. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, pages 1483–1492. SBC.
- Chilana, P. K., Ko, A. J., Wobbrock, J. O., e Grossman, T. (2013). A multi-site field study of crowdsourced contextual help: Usage and perspectives of end users and software teams. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, page 217–226, New York, NY, USA. Association for Computing Machinery.
- Lester, J. C., Converse, S. A., Kahler, S. E., Barlow, S. T., Stone, B. A., e Bhogal, R. S. (1997). The persona effect: affective impact of animated pedagogical agents. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, pages 359–366.
- Marwan, S., Dombe, A., e Price, T. W. (2020). Unproductive help-seeking in programming: What it is and how to address it. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '20, page 54–60, New York, NY, USA. Association for Computing Machinery.
- McKITA, M. (1988). Online documentation and hypermedia: designing learnability into the product. In *Proceedings of the IPCC '88 Conference Record 'On the Edge: A Pacific Rim Conference on Professional Technical Communication'*, pages 301–305, Seattle, WA, USA. IEEE.
- Neto, V. (2013). A utilização da ferramenta scratch como auxílio na aprendizagem de lógica de programação. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação - CBIE 2013*, volume 2.
- Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, page 152–158, New York, NY, USA. Association for Computing Machinery.
- Novick, D. G., Andrade, O. D., e Bean, N. (2009). The micro-structure of use of help. In *Proceedings of the 27th ACM International Conference on Design of Communication*, SIGDOC '09, page 97–104, New York, NY, USA. Association for Computing Machinery.
- Patrick, A. e McGurgan, A. (1993). One proven methodology for designing robust online help systems. In *Proceedings of the 11th Annual International Conference on Systems*

- Documentation*, SIGDOC '93, page 223–232, New York, NY, USA. Association for Computing Machinery.
- Price, T. W., Liu, Z., Cateté, V., e Barnes, T. (2017). Factors influencing students' help-seeking behavior while programming with human and computer tutors. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*, ICER '17, page 127–135, New York, NY, USA. Association for Computing Machinery.
- Ribeiro, R. C. (2019). A utilização do scratch como ferramenta de apoio para suprir as dificuldades dos estudantes de ensino médio no estudo do fenômeno de refração. *Revista Brasileira de Educação, Cultura e Linguagem - RBECL*, 3(6).
- Sales, A. B. d. e Queiroz Palmeira, E. G. (2019). Use preferences in help documentation. In *Proceedings of the 14th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6, Coimbra, Portugal. IEEE.
- Souza, F., Falcão, T., e Mello, R. (2021). O ensino de programação na educação básica: Uma revisão da literatura. In *Anais do XXXII Simpósio Brasileiro de Informática na Educação*, pages 1265–1275, Porto Alegre, RS, Brasil. SBC.
- Voulligny, L. e Robert, J.-M. (2005). Online help system design based on the situated action theory. In *Proceedings of the 2005 Latin American Conference on Human-Computer Interaction*, CLIHC '05, page 64–75, New York, NY, USA. Association for Computing Machinery.
- Welty, C. J. (2011). Usage of and satisfaction with online help vs. search engines for aid in software use. In *Proceedings of the 29th ACM International Conference on Design of Communication*, SIGDOC '11, page 203–210, New York, NY, USA. Association for Computing Machinery.
- Wiggins, J. B., Fahid, F. M., Emerson, A., Hinckle, M., Smith, A., Boyer, K. E., Mott, B., Wiebe, E., e Lester, J. (2021). Exploring novice programmers' hint requests in an intelligent block-based coding environment. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, SIGCSE '21, page 52–58, New York, NY, USA. Association for Computing Machinery.