

# FURBOT-WEB: UMA PLATAFORMA ADAPTATIVA PARA O ENSINO DE PROGRAMAÇÃO

Maurício Capobianco Lopes<sup>1</sup>

Heloisa Kaestner Kopsch<sup>2</sup>

## RESUMO

Este artigo descreve o sistema web nomeado FURBOT-WEB, que tem por objetivo ser uma plataforma de ensino personalizado, a qual permite a programação com blocos. A plataforma de ensino personalizado recomenda exercícios de lógica de programação com base no conhecimento do aluno em determinado conteúdo, utilizando um sistema de recomendação com filtragem por conteúdo. O artigo apresenta o FURBOT\_WEB e apresenta os resultados de sua aplicação em três turmas de estudantes de programação, com base em estatística descritiva. Os resultados obtidos a partir dos experimentos de usabilidade demonstram que o sistema web possui uma interface amigável, atrativa e de fácil utilização, mas que devido a programação com blocos, deixa de ser interessante para os estudantes que já tiveram contato com a programação escrita. A partir dos resultados também se comprovou que a plataforma de ensino personalizado recomendou corretamente os exercícios e é considerada pelos estudantes um recurso importante no aprendizado deles. O trabalho é relevante no âmbito acadêmico, pois em relação ao ensino, automatiza para os professores o processo de aplicação de exercícios e, em relação ao aprendizado, permite aos alunos uma aprendizagem personalizada, atrativa e com potencial de desenvolver melhor suas capacidades.

Palavras-chave: Ensino de Programação. Ensino personalizado. Programação com blocos. Sistema de recomendação. Furbot.

## 1 INTRODUÇÃO

O ser humano está cada vez mais conectado às tecnologias digitais, tendo mais facilidade em adquirir conhecimentos e buscando meios inovadores para isso. Isso gera a preocupação de oferecer recursos didáticos mais atrativos aos alunos das instituições de ensino. Com essa preocupação em mente, propôs-se um ambiente de apoio ao ensino de lógica de programação chamado FURBOT (MATTOS et al., 2008). O ambiente é *desktop* e propõe introduzir a programação para os estudantes de forma mais atrativa. No FURBOT, um

---

<sup>1</sup> Doutor em Engenharia e Gestão do Conhecimento, professor do Programa de Pós-Graduação em Ensino de Ciências Naturais e Matemática, Universidade Regional de Blumenau (FURB).

robô anda por um mundo com aliens, tesouros, números e outros objetos. Os estudantes devem solucionar os problemas propostos utilizando os comandos do próprio FURBOT, além de poderem executar sua resposta e observar se atingiram o resultado esperado.

Uma das limitações do FURBOT é que os alunos recebem os mesmos exercícios, independente do seu nível de conhecimento. Segundo Bacich, Tanzi Neto e Trevisiani (2015, p. 51), “Estudantes da mesma idade não têm as mesmas necessidades, possuem relações diferentes com professores e/ou tecnologias digitais e nem sempre aprendem do mesmo jeito e ao mesmo tempo”. Esta é a base da personalização do ensino, pois os mesmos exercícios para todos os alunos na sala de aula não geram o mesmo aprendizado que exercícios específicos para níveis de conhecimento distintos que podem ser direcionados individualmente para cada aluno. Entretanto, automatizar esse processo de personalização não é uma tarefa simples pois acompanhar individualmente cada aluno requer tempo e recursos apropriados. Neste contexto, as plataformas adaptativas podem auxiliar. Segundo Lopes (2015, p. 1), plataformas adaptativas “São plataformas inteligentes que usam softwares que propõem atividades diferentes para cada aluno, sob medida, a partir de suas respostas e reações às tarefas”. Uma das técnicas conhecidas para implementar plataformas adaptativas são os sistemas de recomendação.

Dada essa problemática, apresenta uma plataforma web adaptativa para o ensino de programação com base no FURBOT utilizando um sistema de recomendação. A plataforma possibilita a individualização do ensino a partir das interações dos alunos com os exercícios disponibilizados através do sistema de recomendação, com base no nível de conhecimento de cada aluno. O software desenvolvido foi testado com estudantes dos cursos de Ciência da Computação e Sistemas de Informação e seus resultados são discutidos no presente artigo.

## **2 REVISÃO BIBLIOGRÁFICA**

Atualmente, o foco da educação ainda está no professor, que detém a informação e possui a responsabilidade sobre a aprendizagem do aluno. O ensino personalizado preocupa-se em mudar essa realidade, permitindo que a responsabilidade da aprendizagem seja redirecionada para o próprio estudante, o qual se torna mais participativo no desenvolvimento de seu conhecimento (BACICH; TANZI NETO; TREVISIANI, 2015). Com isso, o professor passa a ser um mediador que detém o papel de apoio e que passa a ser mais eficiente, pois

---

<sup>2</sup> Graduada em Ciência da Computação, Universidade Regional de Blumenau (FURB)

pode focar em dificuldades específicas que o próprio aluno percebe durante seu aprendizado personalizado. No entanto, existe o desafio em relação aos recursos tecnológicos necessários, como computadores e internet. Uma das preocupações é se todos os alunos terão fácil acesso a esses recursos.

Em um ensino não personalizado o aluno recebe sempre os mesmos conteúdos que os outros alunos, desconsiderando suas individualidades como o nível de conhecimento, as habilidades cognitivas e os seus interesses. Plataformas adaptativas de ensino são desenvolvidas justamente para inverter esse cenário, tendo como principal objetivo disponibilizar ferramentas de ensino que se adaptem às características individuais de cada aluno (GRAF; KINSHUK; IVES, 2010). Uma das técnicas utilizadas em plataformas adaptativas de ensino são os sistemas de recomendação que, segundo Lichtnow et al. (2006, p. 50), têm por objetivo “que os usuários não apenas recebam o retorno de itens a partir da formulação de consultas, mas que o possível interesse do usuário por um determinado item possa ser previsto”.

Uma das técnicas utilizadas neste tipo de sistema é a filtragem por conteúdo que utiliza como base as escolhas passadas do usuário (CAZELLA et al., 2010). Cazella et al. (2010) afirmam que uma técnica comum nesse tipo de abordagem é a indexação de frequência de termos. Já a filtragem colaborativa utiliza como base as escolhas dos usuários que tenham perfis semelhantes ao usuário em questão (CAZELLA et al., 2010). Um exemplo é o sistema recomendar itens que tenham sido bem avaliados por usuários com interesses em comum.

A personalização pode ser aplicada no ensino de programação, uma vez que neste contexto, o nível das turmas em geral é bastante heterogêneo. O aprendizado de programação desenvolve capacidades cognitivas que contribuem para melhorar o raciocínio lógico, pois envolve a habilidade de desenvolver uma solução para um problema (SCAICO, 2013). No trabalho de Rodrigues et al. (2015) é apresentada a correlação entre o desempenho de estudantes no ENEM e a experiência destes em programação. Segundo Rodrigues et al. (2015, p. 129), “alunos com experiência em programação obtiveram um desempenho melhor em todos os eixos cognitivos abordados no exame em relação aos que não estudaram programação”. Além disso, foi observado que “estudantes proficientes em programação possuem desempenho superior de até 13,49% em relação àqueles com pouco ou nenhum nível de experiência”, o que destaca mais um dos benefícios em aprender a programar.

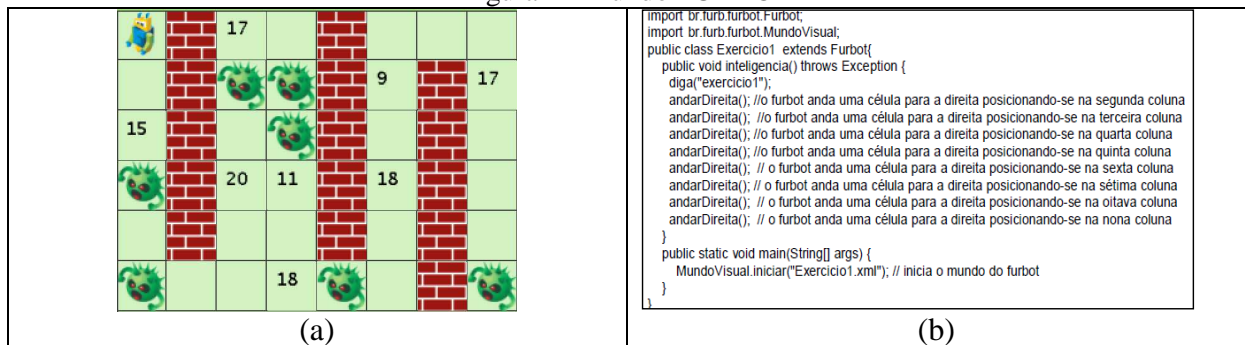
Também na educação básica a importância do ensino de programação tem sido reconhecida por grandes organizações. Como exemplo, destaca-se a Google que tem investido

em projetos na divisão Google na Educação, desenvolvidos para incentivar a introdução de conceitos da Ciência da Computação no ensino médio, e a Hora do Código, um projeto da Code.Org que tem como objetivo desmistificar a ideia de que programação é difícil, introduzindo a programação de forma atrativa através de tutoriais de ensino (CODE.ORG, 2015a).

### 3 METODOLOGIA

A presente pesquisa constitui-se de duas partes: a primeira trata do desenvolvimento do software FURBOT-WEB e a segunda de sua aplicação e análise dos resultados. Como descrito anteriormente, o FURBOT-WEB foi criado com base no FURBOT (MATTOS et al., 2008). Segundo Vahldick e Mattos (2009, p. 3), “o elemento central do FURBOT é a programação de um robô que vive num mundo bidimensional junto de outros tipos de objetos, que também podem ser programados”. A Figura 1(a) apresenta um exemplo do mundo FURBOT no qual é possível ver o robô e alguns objetos como parede, aliens e números.

Figura 1 - Mundo FURBOT



Fonte: digitalizado pelos autores.

A Figura 1(b) apresenta a resolução de um exercício que ocorre através da implementação do método **inteligencia** no qual devem ser definidos os movimentos e ações do robô, utilizando comandos em português expressos na linguagem Java. Como já apresentado, uma das limitações do FURBOT é que ele não utiliza o conceito de personalização de ensino e, por não possuir um controle de perfil por usuário, também não permite uma boa aplicação desse conceito. Com base nisso, desenvolveu-se o software também mudando sua concepção de programação para um sistema baseado em blocos, utilizando a ferramenta Blockly, da Google (GOOGLE, 2015a).

A validação do software foi realizada com base em testes de usabilidade realizado com 69 alunos dentre os quais 26 estavam na disciplina de Introdução a Programação do curso de Sistemas da Informação e os demais estavam na disciplina de Introdução a Programação do

curso de Ciência da Computação da FURB, sendo 22 da turma A e 21 da turma B. Os resultados foram analisados com base em estatística descritiva.

### 3.1.FURBOT WEB

O FURBOT-WEB tem três perfis de usuário: Administrador, Professor e Estudante. As principais funções tratam de manter o cadastro de usuários (papel do Administrador), criar e editar conteúdos, os quais são compostos de exercícios e *links* para materiais de auxílio (papel do Professor) e resolver os exercícios recomendados com base em seu conhecimento (papel do Estudante). No cadastro de conteúdo feito pelo professor é obrigatório associar no mínimo três exercícios. Com os conteúdos e exercícios cadastrados, o professor deve criar uma turma para convidar os alunos para se cadastrarem no sistema web, mas para isso é preciso primeiro cadastrar as perguntas para um quiz utilizado no sistema de personalização.

Para o estudante usar o FURBOT-WEB é necessário que ele aceite o convite que recebeu por e-mail para participar da turma. Feito isso, o aluno é redirecionado para a tela de cadastro na qual ele informa seus dados e responde todas as perguntas do quiz. Por fim, o estudante é redirecionado para a tela inicial do acesso de aluno, que é a tela de resolução de exercício (Figura 2).

Figura 2 - Tela de resolução de exercício

The screenshot displays the 'FURBOT WEB - Ciência da Computação - FURB' interface. At the top, there are tabs for 'Exercícios' and 'Cadastro'. Below the tabs, the 'RESOLVER EXERCÍCIO' section contains a series of numbered steps: 1. Solicitar auxílio para colegas, 2. Visualizar material de apoio, 3. Visualizar código gerado, 4. Executar, 5. Parar execução, 6. Reiniciar mundo, and 7. Encaminhar para correção. A text box explains the robot's movement logic: 'O robô passa pelo modo tradicional. Em cada linha ele soma os valores dos números, sempre iniciando em 0. Sempre que ele chegar ao final da linha deve informar o valor total. No final ele precisa falar qual foi a linha que teve a maior soma. Atenção: não podem ser utilizados atributos. Somente variáveis locais.' Below this is a 10x7 grid of numbers. A central panel lists 'Comandos FURBOT' (Movimentação, Comunicação, Avaliação do mundo, etc.) and 'Comandos Básicos' (Laço, Texto, Lógica, etc.). On the right, a Scratch-style block editor is used to program the robot's path, with blocks for 'definir maiorLinha para 0', 'repetir enquanto não ehFim ABAIXO', 'movimentarRobo com DIRECAO DIREITA', 'andarAbaixo', 'movimentarRobo com DIRECAO ESQUERDA', 'se não ehFim ABAIXO', 'andarAbaixo', 'diga criar texto com <math>+ maiorLinha </math>', 'para movimentarRobo com DIRECAO', 'definir VTotal para 0', 'repetir enquanto não ehFim DIRECAO', 'verificarValor', 'se DIRECAO = DIREITA', 'andarDireita', 'senão andarEsquerda', 'verificarValor', 'diga VTotal', 'se VTotal > maiorVTotal', and 'definir maiorVTotal = VTotal'.

Fonte: digitalizado pelos autores.

Conforme o estudante resolve os exercícios, o professor tem a opção de acompanhar seu progresso em cada um dos conteúdos existentes (Figura 3).

Figura 3 - Tela de acompanhamento do progresso dos alunos

	Básicos	Repetição (ENQUANTO)	Seleção (SE)	Subprogramas	Repetição (PARA)
Fulano da Silva	Green	Green	Green	Green	Red
Fulano da Costa	Grey	Green	Red	Grey	Grey
Fulano dos Santos	Grey	Grey	Grey	Green	Red
Fulano Pereira	Grey	Grey	Grey	Green	Green
Fulano Junior	Green	Green	Green	Green	Green
Fulano Abreu	Grey	Grey	Grey	Green	Red
Fulano Filho	Grey	Grey	Grey	Green	Red
Fulano Neto	Grey	Grey	Grey	Green	Red
Fulano Bisneto	Grey	Grey	Grey	Green	Red

Fonte: digitalizado pelos autores.

Outra opção de funcionalidade do professor é a correção de exercícios. O aluno tem a opção de visualizar essas correções pela tela de consulta de exercícios resolvidos. Quando volta ao sistema, este analisa o perfil e os exercícios resolvidos corretamente pelo estudante e indica novos exercícios a serem feitos.

#### 4 ANÁLISE DOS RESULTADOS

O experimento de usabilidade da experiência do usuário ocorreu no mês de maio por meio de testes coletivos em cada uma das três turmas citadas anteriormente. Para cada uma das turmas foi feita uma apresentação sobre o FURBOT-WEB e seus objetivos. Todos os usuários receberam a tarefa de criar seu cadastro no sistema web e resolver no mínimo um exercício, sendo que o primeiro exercício estava fixo para todos os alunos da turma a fim de manter homogênea a análise dos resultados. Os enunciados dos exercícios foram escolhidos de uma lista de exercícios do FURBOT original, os quais já haviam sido aplicados pelos professores para as três turmas. Após a resolução do exercício fixo, novos exercícios foram recomendados e os estudantes poderiam resolvê-los ou não. Ao fim da aplicação do teste, eles responderam um questionário de usabilidade personalizado para o FURBOT-WEB.

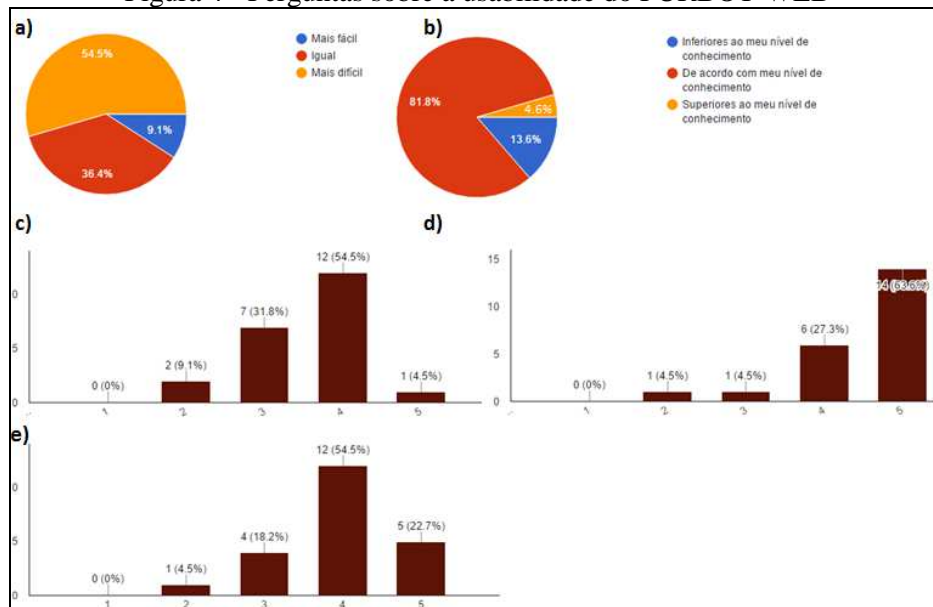
##### 4.1 TURMA A DE CIÊNCIA DA COMPUTAÇÃO

A turma A foi a primeira das três turmas a testar a aplicação. O exercício fixo escolhido para essa turma foi mais complexo em relação aos das outras duas turmas, pois exigia uma quantidade significativa de blocos para efetuar a resolução. Quanto ao perfil da turma, é possível afirmar que a maioria dos alunos dessa turma são do sexo masculino e

possuem idade entre quinze e vinte anos. A metade dos alunos tem experiência com programação entre um e seis meses e todos os alunos possuem acesso à internet em casa.

Na Figura 4 estão representadas graficamente as respostas das cinco próximas perguntas, as quais tratam sobre a usabilidade do FURBOT-WEB. As perguntas estão descritas a seguir, sendo que os itens c e d receberam respostas considerando que o valor um equivale a “discordo totalmente” e o valor cinco equivale a “concordo totalmente” e o item e recebeu respostas considerando que o valor um equivale a “insatisfatório” e o valor cinco equivale a “ótimo”: a) como você avalia a programação com blocos em comparação à programação escrita? b) como você avalia os exercícios recomendados em relação ao seu nível de conhecimento? c) você concorda que, de modo geral, o FURBOT-WEB é de fácil utilização? d) você concorda que a recomendação de exercícios com base no seu conhecimento é importante? e) como você avalia o FURBOT-WEB de forma geral?

Figura 4 - Perguntas sobre a usabilidade do FURBOT-WEB



Fonte: resultados da pesquisa.

Com essas respostas é possível afirmar que a maioria dos alunos considera a programação com blocos mais difícil que a programação escrita e que a maioria dos alunos considerou os exercícios recomendados com complexidade de acordo com o nível de conhecimento deles. A maioria dos alunos concorda parcialmente que o FURBOT-WEB é de fácil utilização, mas concorda totalmente que a recomendação de exercícios com base no conhecimento é importante e que, de forma geral, o FURBOT-WEB é bom. No Quadro 1 estão as dificuldades mais citadas quanto ao uso do FURBOT-WEB e no Quadro 2 as sugestões mais citadas.

Quadro 1 - Dificuldades mais citadas pelos alunos da turma A

Dificuldade	Porcentagem de alunos que citaram
Localizar os blocos correspondentes aos comandos.	40%
Pouco tempo para se adaptar à usabilidade.	18%
Criar variáveis e utilizá-las.	13%

Fonte: resultados da pesquisa.

Quadro 2 - Sugestões mais citadas pelos alunos da turma A

Sugestão	Porcentagem de alunos que citaram
Help da utilização dos comandos.	13%
Organizar melhor a localização dos comandos.	9%
Implementar bloco de comando switch.	9%
Atalho para desfazer ação (ctrl + z).	9%

Fonte: resultados da pesquisa.

A principal dificuldade encontrada pelo professor e alunos da turma na solução do exercício proposto foi que não era possível utilizar as constantes de direção da forma que os alunos estavam acostumados a utilizar no FURBOT original. Assim, as constantes de direção foram implementadas antes dos experimentos de usabilidade das outras turmas.

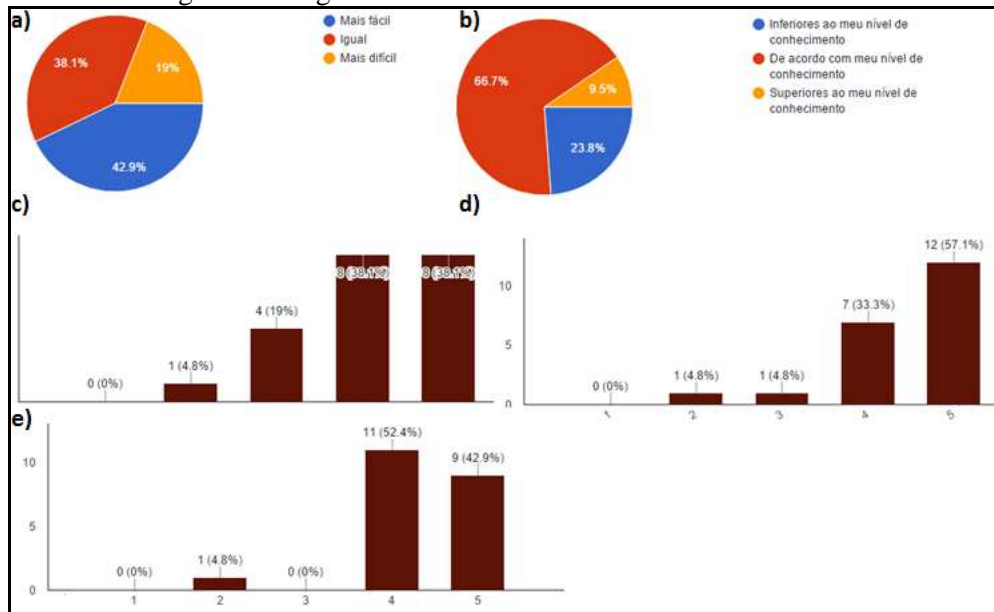
#### 4.2 TURMA B DE CIÊNCIA DA COMPUTAÇÃO

A turma B foi a segunda das três turmas a testar a aplicação. O exercício fixo escolhido para essa turma B foi mais simples que o escolhido para a turma A. Quanto ao perfil, assim como os alunos da turma A, a maioria dos alunos são do sexo masculino e possuem idade entre quinze e vinte anos. Diferente da turma A, a maior quantidade de respostas em relação a experiência com programação, foi da categoria sem experiência e ninguém respondeu que possui mais de três anos de experiência. Todos os alunos possuem acesso à internet em casa.

Na Figura 5 estão representadas as respostas das mesmas questões sobre usabilidade apresentadas à turma anterior.



Figura 5 - Perguntas sobre a usabilidade do FURBOT-WEB



Fonte: resultados da pesquisa.

Diferente dos alunos da turma A, a maior parte dos alunos da turma B considera a programação com blocos mais fácil que a programação escrita. A maioria dos alunos considerou que os exercícios recomendados estavam de acordo com o nível de conhecimento deles e a maioria dos alunos concorda (parcial ou totalmente) que o FURBOT-WEB é de fácil utilização. Em relação a recomendação de exercícios com base no conhecimento, a maioria dos alunos concordou totalmente que é importante. Assim como na turma A, o FURBOT-WEB foi considerado pela maioria, de forma geral, como bom. Se somar as respostas bom e ótimo a aprovação do FURBOT-WEB superou os 95%. No Quadro 3 estão as dificuldades mais citadas e no Fonte: resultados da pesquisa.

Quadro 4 as sugestões mais citadas.

Quadro 3 - Dificuldades mais citadas pelos alunos da turma B

Dificuldade	Porcentagem de alunos que citaram
Localizar os blocos correspondentes aos comandos.	28%
Lógica.	14%
Encaixar os blocos.	14%
Pouco tempo para se adaptar à usabilidade.	14%
Lentidão da execução dos blocos.	14%

Fonte: resultados da pesquisa.

Quadro 4 - Sugestões mais citadas pelos alunos da turma B

Sugestão	Porcentagem de alunos que citaram
Atalho para desfazer ação (ctrl + z).	28%
Permitir resolver o exercício digitando o código fonte.	23%
Corrigir a lentidão da execução.	9%

Fonte: resultados da pesquisa.

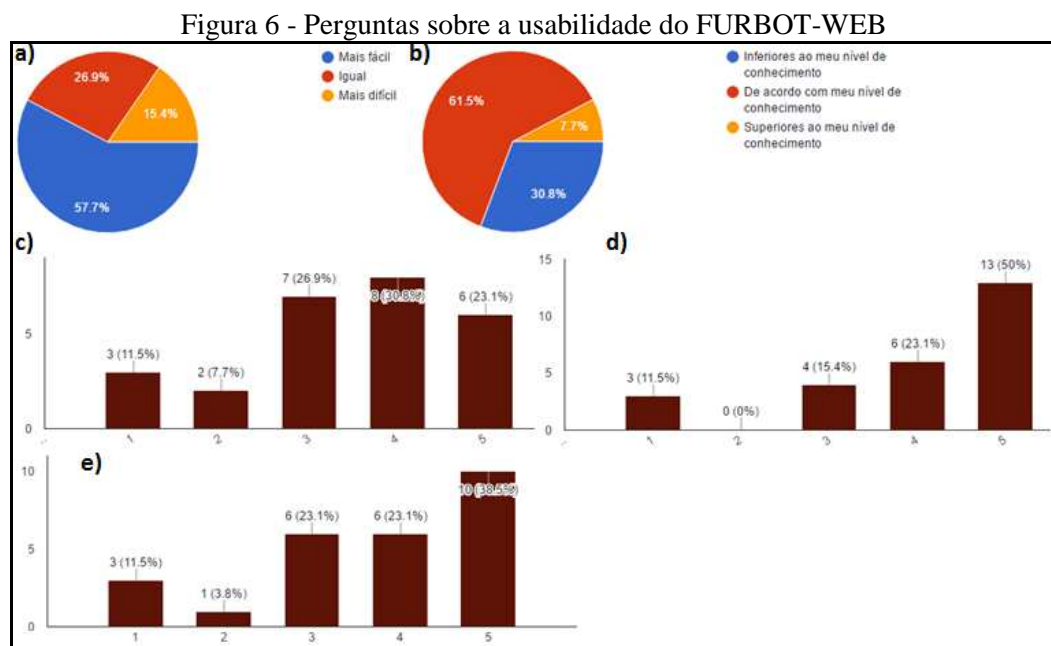
O problema da lentidão da execução do código fonte está relacionado à quantidade de laços utilizados na resolução do exercício e foi identificado antes da aplicação do FURBOT-

WEB nas salas de aula, mas como é uma limitação da biblioteca utilizada, não foi possível corrigir em tempo hábil.

#### 4.2.1.1 Turma de Sistemas da Informação

A turma de Sistemas da Informação (SIS) foi a última das três turmas a testar a aplicação. O exercício fixo escolhido para os alunos de SIS foi o mais simples em relação aos das outras duas turmas, pois se notou que poucos alunos das outras turmas conseguiram finalizar seus exercícios. Quanto ao perfil da turma, assim como os alunos das outras duas turmas, a maioria dos alunos da turma de SIS são do sexo masculino e possuem idade entre quinze e vinte anos. Também a maioria se considera sem experiência com programação ou com no máximo seis meses de experiência. Diferente das outras turmas, quatro alunos de SIS não possuem acesso à internet.

Na Figura 6 estão representadas graficamente as respostas sobre usabilidade do FURBOT-WEB.



Fonte: resultados da pesquisa.

Ao contrário das turmas anteriores, a maioria dos alunos de SIS consideraram a programação com blocos mais fácil que a programação escrita e também que os exercícios recomendados estavam de acordo com o seu nível de conhecimento. A maioria dos alunos concordou que, de modo geral, o FURBOT-WEB é de fácil utilização e concordaram totalmente que a recomendação de exercícios com base em seus conhecimentos é importante.

Diferente das outras turmas, a maioria dos alunos de SIS avaliou o FURBOT-WEB como ótimo. No Quadro 5 estão as dificuldades e no Quadro 6 as sugestões mais citadas.

Quadro 5 - Dificuldades mais citadas dos alunos da turma de SIS

Dificuldade	Porcentagem de alunos que citaram
Pouco tempo para se adaptar à usabilidade.	19%
Localizar os blocos correspondentes aos comandos.	15%
Falta do atalho para desfazer ação (ctrl + z).	7%

Fonte: resultados da pesquisa.

Quadro 6 - Sugestões mais citadas dos alunos da turma de SIS

Sugestão	Porcentagem de alunos que citaram
Atalho para desfazer ação (ctrl + z).	15%

Fonte: resultados da pesquisa.

A falta do atalho para desfazer a ação foi identificada pelas turmas, pois quando tentavam excluir um comando que possuía blocos conectados à ele, todos os blocos também eram excluídos, o que não era a intenção. Em sala de aula poderia ser prevenida essa dificuldade explicando que para excluir um bloco ele deve ser antes desconectado dos blocos que o aluno não pretende excluir.

## 5 CONCLUSÕES

A presente pesquisa procurou identificar contribuições para o ensino-aprendizagem de programação. Para os professores permite automatizar o processo de aplicação de exercícios, o que lhes permite dedicar seu tempo para outras atividades importantes de ensino, como o acompanhamento de alunos que não evoluem nos conteúdos dos exercícios. Para os alunos, o sistema lhes permite uma aprendizagem personalizada, atrativa e com potencial de desenvolver melhor suas capacidades. Também percebeu-se que o FURBOT-WEB poderia ser útil para o ensino de programação para crianças por dois motivos: primeiro por permitir programação com blocos, que demonstrou ser mais fácil que a programação escrita; segundo por possuir uma usabilidade amigável e atrativa do ponto de vista da experiência do usuário, conforme os resultados da pesquisa. O recurso da programação em blocos torna o aprendizado mais atrativo quando os estudantes ainda não entraram em contato com a programação escrita, pois no experimento de usabilidade a maioria das respostas em relação a facilidade da programação com blocos foi de que é mais fácil programar com blocos do que digitando o código fonte.

Em relação ao sistema de personalização destaca-se que a plataforma foi implementada utilizando um sistema de recomendação com filtragem por conteúdo e permite que os

estudantes resolvam exercícios recomendados com base no conhecimento deles, o que não é possível no FURBOT original devido a falta de controle de seção dos usuários. Através dos experimentos de usabilidade verificou-se que a maioria dos estudantes considerou a recomendação de exercícios importante para o aprendizado deles e que a plataforma recomendou exercícios de acordo com o conhecimento dos estudantes. Os resultados permitem inferir o potencial da ferramenta para o ensino de programação.

## REFERÊNCIAS BIBLIOGRÁFICAS

BACICH, Lilian; TANZI NETO, Adolfo; TREVISIANI, Fernando de Mello. **Ensino híbrido: personalização e tecnologia na educação**. Porto Alegre: Penso, 2015.

CAZELLA et al. A Ciência da Opinião: Estado da arte em Sistemas de Recomendação. In: JORNADA DE ATUALIZAÇÃO EM INFORMÁTICA DA SBC, 1, 2010, Rio de Janeiro. **Proceedings...** Rio de Janeiro: JAI, 2010. p. 161-216.

CODE.ORG. **Programação com Anna e Elsa**. [S.I], 2015a. Disponível em: <<https://studio.code.org/s/frozen/stage/1/puzzle/1>>. Acesso em: 09 set. 2015.

GOOGLE. **Blockly: FAQ**. [S.I], 2015a. Disponível em: <<https://developers.google.com/blockly/about/faq>>. Acesso em: 09 set. 2015.

GRAF, Sabine; KINSHUK; IVES, Cindy. A flexible mechanism for providing adaptivity based on learning styles in learning management systems. In: INTERNATIONAL CONFERENCE ON ADVANCED LEARNING TECHNOLOGIES, 10, 2010, Sousse. **Proceedings...**Sousse: ICALT, 2010.

LICHTNOW, Daniel et al. O Uso de Técnicas de Recomendação em um Sistema para Apoio à Aprendizagem Colaborativa. **Revista Brasileira de Informática na Educação**, Porto Alegre, v. 14, n. 3, p. 50-59, set. 2006.

LOPES, Marina. **Plataformas adaptativas**. São Paulo, 2015. Disponível em: <<http://porvir.org/plataforma-adaptativa/>>. Acesso em: 16 jun. 2016.

MATTOS, Mauro M. et al. **Apostila de OO e FURBOT**. Blumenau, 2008. Disponível em: <[http://www.inf.furb.br/poo/furbot/files/Apostila\\_FURBOT.pdf](http://www.inf.furb.br/poo/furbot/files/Apostila_FURBOT.pdf)>. Acesso em: 30 ago. 2015.

RODRIGUES, Rivanilson S. et al. Análise dos efeitos do Pensamento Computacional nas habilidades de estudantes no ensino básico: um estudo sob a perspectiva da programação de computadores. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 26, 2015, Maceió. **Anais...** Maceió: SBIE, 2015. p. 121-130. Disponível em: <<http://www.br-ie.org/pub/index.php/sbie/article/view/5125/3529>>. Acesso em: 13 de mar. 2016.

SCAICO, Pasqueline D. et al. Ensino de Programação no Ensino Médio: Uma Abordagem Orientada ao Design com a linguagem Scratch. **Revista Brasileira de Informática na Educação**, Porto Alegre, v. 21, n. 2, p. 92, set. 2013.

VAHLDICK, Adilson; MATTOS, Mauro M. Aprendendo programação de computadores com experiências lúdicas. In: INTERNATIONAL CONFERENCE ON ENGINEERING AND COMPUTER EDUCATION, 6, 2009, Buenos Aires. **Anais...** Buenos Aires: ICECE, 2009. Não paginado. Disponível em:  
<<http://sites.google.com/site/adilsonv77/icece09.pdf?attredirects=0>>. Acesso em: 10 set. 2015.

**Recebido em abril 2018**  
**Aprovado em junho 2018**