Um Mapeamento Sistemático sobre Iniciativas Brasileiras em Ambientes de Ensino de Programação

Priscylla Silva[1], Maria Cristina Tenório[2], Joseana Fechine[1], Evandro Costa[2]

[1]Universidade Federal de Campina Grande (UFCG) – Campina Grande, PB – Brazil

[2]Instituto de Computação – Universidade Federal de Alagoas (UFAL)

priscyllamaria@copin.ufcg.edu.br, mctcc@ic.ufal.br

joseana@dsc.ufcg.edu.br, evandro@ic.ufal.br

Abstract. *This paper presents a study on the Brazilian initiatives regarding the creation of programming learning environments. It was performed a Systematic Mapping from main Brazilian events in the area: CBIE, SBIE, WIE and WEI. The mapping also included the Brazilian journal RBIE. Among 228 primary studies, 22 were included in this study. The results show that there was an increase of publications on creating programming learning environments, although there is little investment in personalization and adaptativity. With this mapping, we want to offer a historical survey of Brazilian initiatives, thus facilitating new studies.*

Resumo. *Este artigo apresenta um estudo sobre iniciativas brasileiras sobre a criação de ambientes de aprendizagem de programação. Foi realizado um Mapeamento Sistemático da Literatura utilizando os principais eventos brasileiros da área: CBIE, SBIE, WIE e WEI. O mapeamento também incluiu a revista RBIE. Entre os 228 estudos primários selecionados, 22 foram incluídos no estudo. Os resultados mostraram que houve um aumento das publicações sobre a criação de ambientes de aprendizagem de programação, embora haja pouco investimento em personalização e adaptatividade. Com esse mapeamento espera-se oferecer um levantamento histórico das iniciativas brasileiras, facilitando novos estudos nessa área.*

1. Introdução

O ensino de programação para iniciantes apresenta uma série de problemas que atingem tanto alunos, quanto professores [Scaico et al. 2013]. No que diz respeito à aprendizagem de programação, a literatura aponta vários fatores para justificar o insucesso dos iniciantes, tais como: falta de habilidade para resolução de problemas, dificuldade em expressar solução de problemas de um modo geral, dificuldade em expressar solução de um problema em uma linguagem de programação, falta de compreensão da estrutura da linguagem, entre outros.

Já os professores precisam lidar com turmas densas, cuja média de alunos matriculados é de aproximadamente 40 alunos. Com o objetivo de estimular o aprendizado, e como método didático comumente utilizado no ensino de programação, esses docentes solicitam aos alunos um número elevado de exercícios, gerando assim sobrecarga para si,

DOI: 10.5753/cbie.sbie.2015.367

visto a ausência de tempo hábil e o caráter custoso em avaliar manualmente o conjunto de respostas gerado pelos alunos para cada grupo de questões [Alves and Jaques 2014].

Pesquisas motivadas pelos problemas supracitados propõem, para sua solução, a criação de sistemas computacionais a fim de apoiar o ensino de programação, chamados de Ambientes de Ensino de Programação. Como exemplo tem-se ambientes como o The Huxley [Paes et al. 2013], NewProg [Torezani et al. 2013], Hoopaloo [do Nascimento et al. 2010], entre outros.

Este artigo apresenta um mapeamento sistemático da literatura baseado nos principais eventos de informática na educação no Brasil com o objetivo de analisar as iniciativas brasileiras de criação de ambientes de ensino de programação.

Para escolha das fontes de pesquisa foram selecionados eventos e periódicos na área de Informática na Educação. Os eventos selecionados foram: Workshops do Congresso Brasileiro de Informática na Educação (WCBIE), Workshop sobre Educação em Computação (WEI), Simpósio Brasileiro de Informática na Educação (SBIE) e Workshop de Informática na Escola (WIE). Em conjunto com esses eventos foi adicionada como fonte a Revista Brasileira de Informática na Educação (RBIE). A escolha das fontes de pesquisa foi baseada na repercussão nacional desses veículos, juntamente com o seu *Qualis* fornecido pela CAPES.

Com este mapeamento foi possível constatar os usuários alvo de tais ambientes, além de apresentar uma visão geral das linguagens de programação abordadas por eles. Também constatou-se que, embora haja nesses ambientes a verificação da corretude da solução dos estudantes, poucos oferecem suporte para sanar as dificuldades ou auxílio na resolução dos exercícios de programação.

2. Metodologia

De acordo com Kitchenham and Charters [Kitchenham and Charters 2007] uma revisão sistemática da literatura é realizada para identificar, avaliar e interpretar os estudos disponíveis na literatura e que são relevantes para determinadas questões de pesquisa. No contexto desse trabalho, foi adotado a revisão sistemática como metodologia para encontrar e compartilhar iniciativas brasileiras de criação de ambientes de ensino de programação, a partir das publicações da RBIE e dos anais do SBIE, WIE, WEI e Workshops do CBIE.

Para realização do mapeamento sistemático, foram utilizadas as propostas de Petersen et al. [Petersen et al. 2008], que dividem o mapeamento em quatro etapas:

- Definição das questões de pesquisa;
- Execução da pesquisa por estudos primários relevantes;
- Extração e síntese dos dados;
- Análise dos dados.

2.1. Definição das Questões de Pesquisa

A questão central que motivou esse mapeamento foi: *Qual o panorama histórico dos ambientes de ensino de programação produzidos no Brasil?*. Para responder a essa questão, foram definidas as seguintes questões específicas de pesquisa:

1. Como ocorreu a evolução histórica das publicações sobre Ambientes de Ensino de Programação produzidos Brasil?
2. Os ambientes analisados possuem soluções para o perfil do estudante? Se sim, como eles utilizam o perfil?
3. Quais linguagens de programação esses ambientes oferecem suporte?
4. Para quais disciplinas acadêmicas esses ambientes foram criados?
5. Quais tipos de ajuda esses ambientes oferecem aos estudantes durante a resolução de exercícios?
6. Que tipo de avaliação de exercícios esses ambientes realizam?
7. Os ambientes propostos são usados em sala de aula?

2.2. Pesquisa de Estudos Primários

Após a definição das questões de pesquisa, foram definidas as fontes. Dado o enfoque brasileiro da pesquisa, foram escolhidos como fontes eventos e periódicos do Brasil voltados para educação e informática. A pesquisa foi realizada de forma manual nos engenhos de busca disponíveis de cada fonte. As fontes de busca selecionadas foram:

- Workshops do Congresso Brasileiro de Informática na Educação - WCBIE (http://www.br-ie.org/pub/index.php/wcbie/index);
- Workshop sobre Educação em Computação - WEI (http://www.lbd.dcc.ufmg.br/bdbcomp/servlet/PesquisaEvento?evento=WEI);
- Revista Brasileira de Informática na Educação - RBIE (http://br-ie.org/pub/index.php/rbie/index);
- Simpósio Brasileiro de Informática na Educação - SBIE (http://br-ie.org/pub/index.php/sbie/);
- Workshop de Informática na Escola - WIE (http://br-ie.org/pub/index.php/wie/index).

O período dos artigos publicados varia em relação às fontes de busca: WCBIE (2012 a 2014), WEI (2000 a 2014), RBIE (1997 a 2014), SBIE (2001 a 2014), WIE (2003 a 2014, com exceção de 2004, por não estar disponível *online*).

A *string* de busca utilizada em todas as fontes foi "**program***". Optou-se por um termo mais geral por ser um radical presente tanto no português (programação) quanto no inglês (programming). Embora muitos dos resultados encontrados não fossem relevantes, foi decidido continuar com o termo para não perder nenhum possível resultado relevante.

Os seguintes critérios de inclusão foram utilizados no mapeamento:

- Conter a *string* de busca no título ou no resumo;
- Possuir versão completa disponível *online*;
- Em caso de artigos que tratavam do mesmo ambinente, apenas o mais recente foi incluído.

Foram utilizados como critérios de exclusão:

- Ambientes cujo foco principal não eram o ensino de programação. Ex: ensino de robótica;
- Artigos sobre mecanismos de ensino de programação que não estavam integrados a ambientes utilizáveis;

- Artigos sobre ambientes que não haviam sido implementados;
- Artigos sobre ambientes que não foram testados ou experimentados;
- Artigos sobre *plugins* para alguma IDE (Integrated Development Environment).

As publicações também foram avaliadas de acordo com sua qualidade. Como Critérios de Qualidade, foi decidido considerar todos os artigos que indicassem claramente o objetivo da criação do ambiente, os seus usuários-alvo e que tivessem realizado no mínimo um estudo piloto como forma de teste.

Inicialmente, usando apenas a *string* de busca, foram encontrados um total de 288 artigos. Foi realizada uma triagem, onde foram aplicados os critérios de inclusão e exclusão para cada um dos artigos recuperados. Após a triagem, foram selecionadas 25 publicações. Entre essas, 3 ambientes eram relatados em dois artigos cada um, por tanto o artigo mais antigo foi excluído. Ao final, foram analisadas 22 publicações. Um lista detalhada de cada artigo está disponível no link: `https://goo.gl/0gmg7M`.

3. Resultados e Discussões

Nessa seção serão abordadas as duas últimas etapas do mapeamento: 3) Extração e síntese dos dados; 4) Análise dos dados. Para cada uma das questões de pesquisa, iremos relatar o sumário dos dados coletados juntamente com sua análise e discussões.

3.1. Questão 1: Como ocorreu a evolução histórica das publicações sobre Ambientes de Ensino de Programação produzidos Brasil?

Na Figura 1 pode-se observar a evolução histórica das publicações sobre Ambientes de Ensino de Programação nos principais veículos brasileiros sobre Informática na Educação. Percebeu-se que houveram dois períodos de tempo em que não ocorreu nenhuma publicação: entre 2000 e 2001 e no período de 2003 a 2005. Além disso, o número de publicações entre 2011 e 2013 se manteve constante, tendo ocorrido um aumento significativo em 2014. Essa estabilidade e aumento, em parte, deve-se a criação do CBIE e abertura dos Workshops com mostras de software e de ambientes virtuais na educação. Esses workshops abriram a possibilidade de publicações e demonstrações de softwares educacionais desenvolvidos nas instituições de ensino.

3.2. Questão 2: Os ambientes analisados possuem soluções para o perfil do estudante? Se sim, como eles utilizam o perfil?

Uma questão de pesquisa importante está relacionada as soluções para o perfil do estudante que esses ambientes possuem. Esta é uma questão relevante pois a existência de um perfil do estudante permite que o ambiente ofereça recursos e mecanismos personalizáveis e adaptativos. Como podemos ver na Figura 2, em sua maioria, os ambientes analisados não possuem soluções para o perfil do estudante.

"Perfil", no contexto desta análise, consiste no armazenamento de informações do estudante que vão além das informações básica de cadastro. Um perfil geralmente mantém informações sobre preferências, nível de conhecimento do estudantes ou assuntos que eles têm dificuldades. Dos 5 ambientes que possuem soluções de perfil, um deles utiliza uma solução estatística e os 4 demais utilizam soluções de "modelo do estudante", essa é uma solução mais sofisticada na qual são utilizadas técnicas de Inteligência Artificial. Algumas soluções para criação do perfil do estudante especificamente no domínio

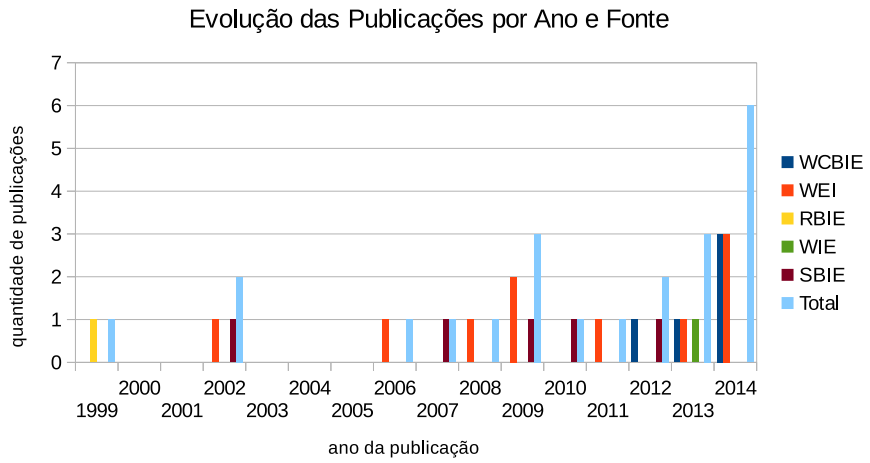

Figura 1. Evolução Histórica das Publicações por Ano e Fonte de Busca.

de programação são apresentadas por Oliveira et al. [Oliveira et al. 2014], Pimentel et al. [Pimentel et al. 2003] e por Gonzales and Tamariz [Gonzalez and Tamariz 2014].

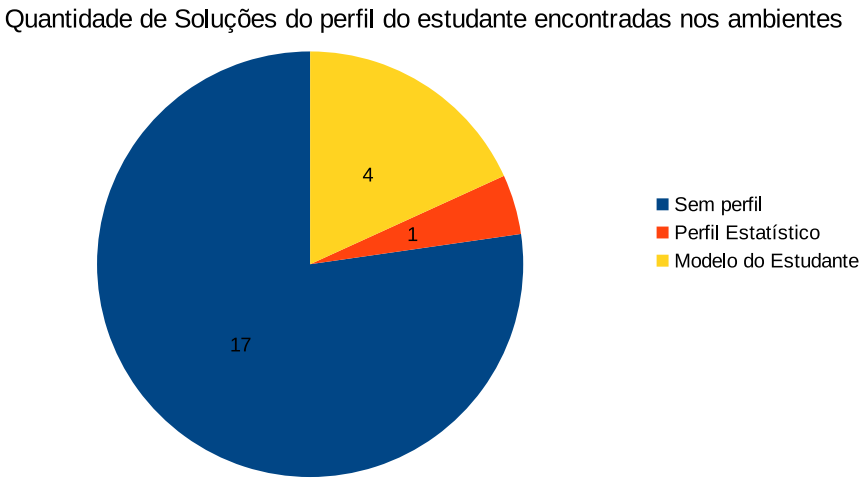

Figura 2. Soluções do perfil do estudante encontradas nos ambientes.

3.3. Questão 3: Quais linguagens de programação esses ambientes oferecem suporte?

No que se refere as linguagens de programação suportadas pelos ambientes, percebe-se que a maioria optou pelo uso de pseudocódigo (ver Figura 3). Essa é uma escolha comum entre professores de disciplinas introdutórias de programação, devido ao nível de abstração e a aproximação com a linguagem usado no dia-a-dia pelos estudantes. Alguns artigos não especificavam as linguagens suportadas pelos ambientes, classificando-o apenas como "multilinguagem", esse foi o caso de três artigos. Os artigos que indicavam quais linguagens o ambiente oferecia suporte foram computados uma vez para cada linguagem. Vale ressaltar que duas das publicações analisadas apresentavam ambientes de ensino de programação que suportavam apenas exercícios com questões objetivas.

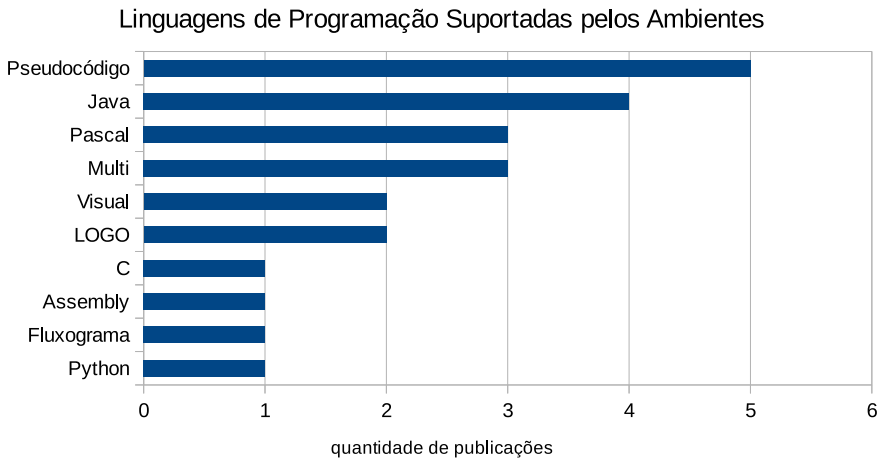

Figura 3. Linguagens de Programação suportadas pelos Ambientes das publicações.

3.4. Questão 4: Para quais disciplinas acadêmicas esses ambientes foram criados?

Ambientes de ensino de programação são criados com um objetivo específico. É possível identificar na maioria das publicações que estes ambientes foram concebidos para uso em disciplinas de Introdução a Programação (ver Figura 4). Esse resultado já era esperado, dada a preocupação da comunidade brasileira e mundial com o alto índice de reprovação dos alunos nessa disciplina. Dentre os ambientes, apenas dois deles eram voltados para iniciantes em programação sem estar vinculado a disciplinas acadêmicas, eles foram criados tendo como usuário-alvo crianças e adolescentes.

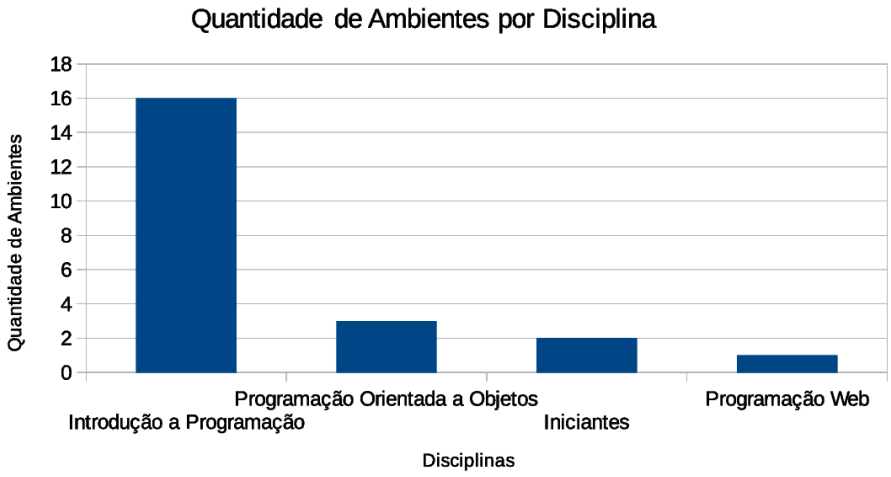

Figura 4. Quantidade de Ambientes criados para cada disciplina acadêmica.

3.5. Questão 5: Quais tipos de ajuda esses ambientes oferecem aos estudantes durante a resolução de exercícios?

Dos artigos analisados, 8 não oferecem nenhum tipo de auxílio para o estudante durante a resolução de exercícios. Na Figura 5 pode-se observar que explicações dos comandos

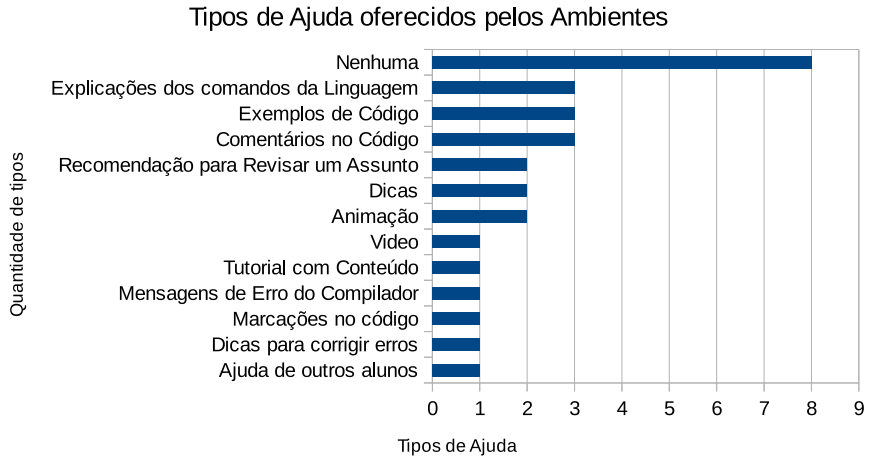

Figura 5. Tipos de Ajuda oferecidos pelos Ambientes.

da linguagem, exemplos de código e comentários no código são os recursos mais comuns nos ambientes. Entretanto os dois primeiros são recursos de ajuda estáticos. E no caso dos comentários, todos são feitos manualmente pelo professor, tutor ou colega de turma. Dos ambientes que oferecem auxílio, 4 são somente de forma automática (controlada pelo ambiente), 3 são de forma automática e também permitem que o estudante peça ajuda, e em 7 ambientes existe apenas a opção para o estudante pedir ajuda ou buscar ajuda em um menu.

3.6. Questão 6: Que tipo de avaliação de exercícios esses ambientes realizam?

Com relação ao tipo de avaliação de exercícios, foi realizada uma verificação de como os ambientes avaliavam e atribuíam nota as soluções algorítmicas dos estudantes. Apenas 8% dos ambientes faziam uma avaliação qualitativa das soluções, os demais utilizavam uma abordagem de testes para avaliar apenas a corretude da solução. No contexto deste trabalho, consideramos avaliações qualitativas aquelas cujo objetivo é verificar determinados atributos que indicam a qualidade do algorítmo, tais como: identação, complexidade, métricas de engenharia de software, legibilidade, entre outros. O Ambiente AIIP [Gomes et al. 2011], por exemplo, utiliza as seguintes métricas para avaliação: linhas de código, variáveis, instruções e blocos de condição e repetição. As avaliações que utilizam a abordagem de testes, de modo geral, fornecem uma entrada de dados ao algorítmo e verificam se as saídas correspondentes estão corretas.

3.7. Questão 7: Os ambientes propostos são usados em sala de aula?

Dos artigos analisados, em apenas 6 deles os ambientes estavam sendo usados no dia-a-dia pelos estudantes das disciplinas (ver Figura 6). A grande maioria (13 ambientes) foi usada apenas em experimentos com um número reduzido de participantes. Em 3 casos, o ambiente havia sido utilizado apenas em um teste piloto. Esta questão é importante, pois seus resultados indicam que poucos dos ambientes desenvolvidos realmente chegaram a ser usados pelos estudantes no decorrer de um semestre completo. Vale ressaltar que dos artigos analisados, apenas 3 ambientes tiveram mais de uma publicação (duas cada um), os demais não voltaram a ser publicados com melhorias nos anos posteriores.

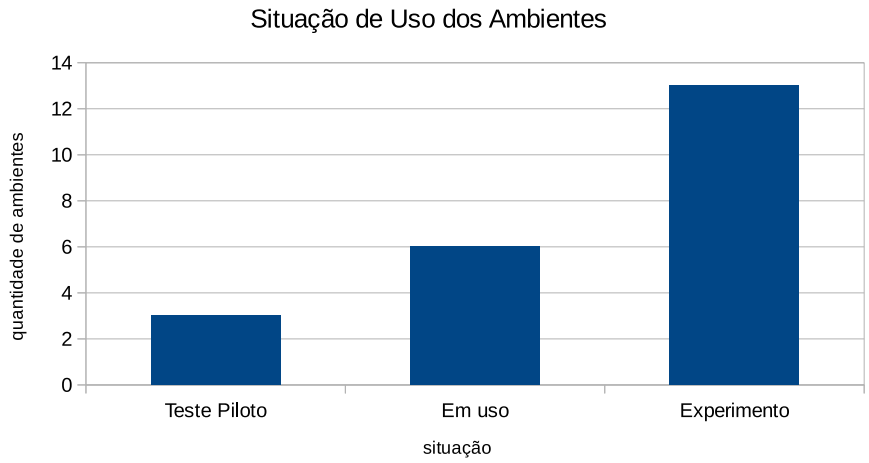

Figura 6. Situação de Uso dos Ambientes Analisados.

4. Ameaças à Validade do Estudo

As principais ameaças à validade do mapeamento sistemático realizado nesse trabalho estão relacionadas a seleção das publicações e a extração de seus dados. De acordo com um dos critérios de inclusão, foram selecionados apenas os trabalhos que possuíam a *string* de busca no título ou resumo/abstract, o que pode ter excluído alguns resultados relevantes. Não foram usados sinônimos na *string* de busca, o que também pode ter causado a exclusão de alguma publicação pertinente. A extração dos dados para sumarização e análise foi realizada manualmente, e embora tenham sido rigorozamente revisadas, existe um pequeno risco de que contenham erros.

5. Considerações Finais e Trabalhos Futuros

O mapeamento sistemático apresentado neste trabalho permitiu analisar as iniciativas brasileiras na criação de Ambientes de Ensino de Programação publicadas nos principais veículos nacionais da área de Informática na Educação. Com a análise dos dados extraídos do mapeamento foi possível responder as questões de pesquisa definidas. Observou-se o aumento de publicações relacionadas a ambientes de ensino de programação no ano de 2014, em parte devido aos Workshops que integram o CBIE. É importante destacar que todos os artigos selecionados foram escritos em Língua Portuguesa.

Outro ponto importante descoberto durante a análise de resultados foi o baixo número de ambientes que de fato chegaram a ser utilizados pelos estudantes durante um ou mais semestres. Além disso, identificou-se um baixo índice de ambientes que ofereciam ajuda personalizada aos estudantes durante a resolução de exercícios, um dos motivos para isto pode ter sido a falta de soluções para a criação de perfis dos estudantes que forneceriam as informações necessárias para a personalização.

Os estudos primários retornados neste trabalho servirão como referência para a construção de um ambiente de ensino de programação que ofereça mecanismos personalizáveis de ajuda ao estudante durante a resolução de exercícios de programação.

Referências

Alves, F. and Jaques, P. (2014). Um ambiente virtual com feedback personalizado para apoio a disciplinas de programação. In *XXV Simpósio Brasileiro de Informática na Educação (SBIE 14)*, Dourados, MS.

do Nascimento, M., Mendonça, A., Guerrero, D., and de Figueiredo, J. (2010). Teaching programming for high school students: A distance education experience. In *Frontiers in Education Conference (FIE), 2010 IEEE*, pages F1J–1–F1J–6.

Gomes, C., Lima, D., Ribeiro, R., Almeira, E., and Brito, P. (2011). Uma proposta para auxiliar alunos e professores no ensino de programação: O ambiente aiip. In *XIX Workshop sobre Educação em ComputaÃ§Ã£o (WEI 11)*, Natal, RN.

Gonzalez, S. and Tamariz, A. (2014). Integração de uma metodologia de ensino presencial de programação com um sistema tutor inteligente. *Revista Brasileira de Informática na Educação*, 22(02):16.

Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering.

Oliveira, M., Monroy, N., Zandonade, E., and Oliveira, E. (2014). Análise de componentes latentes da aprendizagem de programação para mapeamento e classificação de perfis. In *XXV Simpósio Brasileiro de Informática na Educação (SBIE 14)*, Dourados, MS.

Paes, R., Malaquias, R., Guimarães, M., and Almeira, H. (2013). Ferramenta para a avaliação de aprendizado de alunos em programação de computadores. In *Anais dos Workshops do CBIE 2013*, Dourados, MS.

Petersen, K., Feldt, R., Mujtaba, S., and Mattsson, M. (2008). Systematic mapping studies in software engineering. In *Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering*, EASE'08, pages 68–77, Swinton, UK, UK. British Computer Society.

Pimentel, E., França, V., Noronha, R., and Omar, N. (2003). Avaliação contínua da aprendizagem, das competencias e habilidades em programação de computadores. In *XXV Workshop de Informática na Escola (WIE 03)*.

Scaico, P., Lima, A., Azevedo, S., Silva, J. B., Raposo, E., Paica, L. F., Alencar, Y., Mendes, J. P., and Scaico, A. (2013). Ensino de programação no ensino médio: Uma abordagem orientada ao design com a linguagem scratch. *Revista Brasileira de Informática na Educação*, 21(02):92.

Torezani, C., Chagas, L., and Tavares, O. (2013). Newprog - um ambiente online para crianças aprenderem programação de computadores. In *XXV Workshop de Informática na Escola (WIE 13)*, Campinas, SP.