

Aprendizagem Ativa no Ensino de Programação: Uma Revisão Sistemática da Literatura

Fernando Lucas de Oliveira Farias e Isabel Dillmann Nunes

Instituto Metrôpole Digital – Universidade Federal do Rio Grande do Norte (UFRN)
Av. Sen. Salgado Filho, 3000 – Lagoa Nova, CEP: 59.078-970 – Natal – RN – Brasil

{fernandoo.mcp, beldillnunes}@gmail.com

Abstract. *Many studies in the field of computing and education present tools or environments related to teach in programming that do not reach the expected effect on the student's learning experience. The empirically associated factor to the non-observance in the construction of the educational scenarios and active teaching methodologies are one of the reasons. In this sense, the paper aims to present the results of a Systematic Literature Review (SLR) about processes, tools and methodologies that assist in the active learning of programming. The SLR initially gathered 363 candidate studies, among them 29 studies were selected according to the protocol, reflecting the state of the art for the researched theme.*

Resumo. *Inúmeros estudos na área de computação e educação apresentam ferramentas ou ambientes para ensino de programação que não surtem o efeito esperado na experiência de aprendizagem do estudante, fator associado empiricamente a inobservância na construção dos cenários educacionais e metodologias ativas de ensino que tornem o aluno protagonista na construção do seu conhecimento. Neste sentido, o trabalho tem como objetivo apresentar os resultados de uma Revisão Sistemática da Literatura (RSL) sobre processos, ferramentas e metodologias que auxiliem na aprendizagem ativa de programação. A RSL reuniu inicialmente 363 estudos candidatos, dentre eles 29 estudos foram selecionados em conformidade ao protocolo, refletindo o estado da arte para temática pesquisada.*

1. Introdução

O desenvolvimento de habilidades transversais e interdisciplinares surge como uma oportunidade ímpar para que o aprendiz seja preparado para profissões que ainda não existem, tornando-o competitivo frente aos processos de automatização, digitalização e robotização que estão extinguindo milhões de empregos ao redor do mundo.

Segundo Stefanova Ratcheva e Till Leopold (2018), autores de um estudo para o Fórum Econômico Mundial, 75 milhões dos empregos atuais irão desaparecer nos próximos quatro anos, enquanto 133 novas profissões irão surgir para atender às novas necessidades da economia global, sugerindo uma lista de 10 habilidades transversais, interdisciplinares e fundamentais que devem ser dominadas por todo profissional até 2022, são elas: 1.pensamento analítico e inovador; 2.aprendizagem ativa e estratégias de aprendizado; 3.criatividade, originalidade e iniciativa;

4.design da tecnologia e programação; 5.pensamento crítico e análise; 6.resolução de problemas complexos; 7.liderança e gestão de pessoas; 8.inteligência emocional; 9.raciocínio lógico, resolução de problemas e ideação; 10.análise de sistemas e avaliação.

A formação de novos profissionais na área de tecnologia, bem como em outras áreas do conhecimento em programação de computadores é uma tarefa desafiadora objeto de centenas de pesquisas científicas, propondo diversas estratégias que abordam desde robótica, linguagem de programação e robótica, jogos educacionais, metodologias ativas de ensino até arquitetura pedagógica; além de abordagens pedagógicas envolvendo aprendizagem significativa, construcionista, instrucionista, taxionomia de bloom e o Ciclo de Kolb [Silva *et al.* 2015].

Todavia, o uso de metodologias ativas de ensino no contexto da aprendizagem de programação ainda surge como obscuro em muitas realidades, principalmente por ainda ocorrer conflitos na postura epistemológica do professor e do estudante, bem como pela adoção de métodos conservadores de avaliação centrados na resolução exaustiva de exercícios por parte do aprendiz que são frágeis quando se observa o desenvolvimento holístico das habilidades relacionadas aos pilares do pensamento computacional premissas ao aprendizado de programação.

Neste contexto, a pesquisa sistemática realizada teve como principal objetivo, selecionar estudos relevantes que apresentam ou propõem processos, ferramentas, ambientes, metodologias ativas de ensino, estudos de casos envolvendo a aprendizagem ativa de programação para estudantes do ensino médio e superior, contribuindo na construção de cenários educacionais inovadores envolvendo o aprendizado ativo de programação.

2. Revisão Sistemática

A revisão sistemática da Literatura (RSL) apresentada neste trabalho, tem como objetivo identificar estudos que propõem ou relatam processos, ferramentas, ambientes, metodologias ativas de ensino ou estudos de caso envolvendo a aprendizagem ativa de programação para estudantes da educação básica e superior.

A aprendizagem ativa no contexto do ensino de programação será considerada para efeitos de análise dos estudos selecionados, àquela apoiada por metodologias ativas de ensino, apresentada por Moran (2018) como sendo atividades ou ambientes em que o estudante percorre suas trilhas de aprendizagem, sem a intervenção constante do professor em seu próprio ritmo ou tempo.

A metodologia utilizada seguiu as três etapas e atividades sugeridas por Kitchenham (2004) que envolvem: Planejamento, condução e divulgação dos resultados ilustradas no infográfico da Figura 1.

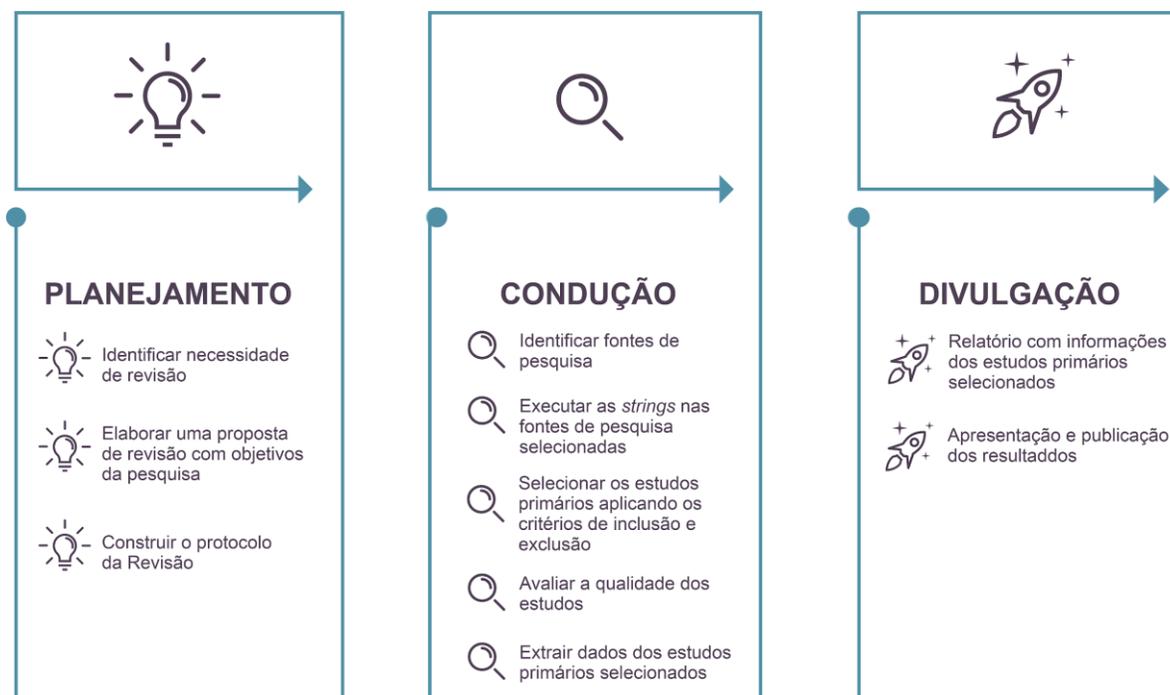


Figura 1. Infográfico com etapas e atividades da RSL

A etapa de planejamento teve início com identificação da necessidade de revisão e definição das diretrizes para levantamento do estado da arte para o trabalho, resultando na elaboração do protocolo¹ da pesquisa, sendo este uma bússola para fase de condução da revisão sistemática da literatura.

Na etapa de condução foram selecionadas bases de pesquisa eletronicamente indexadas, com acesso irrestrito e suporte a strings de busca com expressões lógicas aplicáveis ao título (*title*), resumo (*abstract*) e palavras-chave (*keywords*) dos trabalhos indexados, conhecidas pela comunidade de pesquisadores da área de ciência da computação e educação, neste sentido, as bases selecionadas foram: *IEEE Xplore*, *Scopus*, *ACM Digital Library*, *CEIE* e *Engineering Village*.

2.1. Questões de Pesquisa

A seleção dos estudos primários foi precedida pela definição das questões de pesquisa norteadoras, importantes a extração de dados dos estudos selecionados e fundamentais para resposta a questão principal de pesquisa “quais os processos, ferramentas, ambientes, metodologias ativas de ensino envolvendo aprendizagem ativa no ensino de programação para estudantes do ensino médio e superior?”, as questões estabelecidas foram:

- QP01: Qual nível de ensino dos estudantes participantes do estudo?
- QP02: Os estudantes são de cursos não relacionados a computação?
- QP03: Quais metodologias ativas de ensino apoiam aprendizagem ativa de programação?

¹ Disponível em: <<http://bit.ly/ppgite-dissertação-rsl-protocolo>> Acesso em: 13 jul. 2019

- QP04: Quais ferramentas são utilizadas para aprendizagem ativa de programação?

Após definição das questões de pesquisa, duas *strings* de busca foram formuladas, sendo uma com termos em português e outra em inglês, por meio de uma expressão lógica para obtenção de estudos candidatos enquadrados no objetivo estabelecido para pesquisa; retornando resultados no âmbito do ensino de programação, aprendizagem ativa de programação, habilidades técnicas relacionadas ao ensino de programação ou ainda metodologias ativas de ensino no aprendizado ativo de programação, as *strings* foram: “*computer programming skills*” OR “*active programming learning*” OR “*teaching computer programming*”; “*habilidades de programação*” OR “*aprendizagem ativa de programação*” OR “*ensino de programação computadores*”.

Para assegurar a efetividade em cada base de pesquisa consultada foram necessários alguns ajustes às *strings* de busca, a fim de que fossem adequadas às especificidades de sintaxe identificadas, neste sentido, identificamos na Tabela 1 uma maior quantidade de estudos candidatos na fase de pré-seleção, retornados pelas bases de pesquisa *engineering village* (163), IEEE Xplore (62) e ACM DL (57).

ID	Fonte	String	Resultados
SS01	Scopus	TITLE-ABS-KEY(“computer programming skills” OR “active programming learning” OR “teaching computer programming”)	44
SS02	Scopus	TITLE-ABS-KEY(“habilidades de programação” OR “aprendizagem ativa de programação” OR “ensino de programação computadores”)	08
SS03	IEEE Xplore	“computer programming skills” OR “active programming learning” OR “teaching computer programming”	62
SS04	IEEE Xplore	“habilidades de programação” OR “aprendizagem ativa de programação” OR “ensino de programação computadores”	00
SS05	CEIE	“computer programming skills” OR “active programming learning” OR “teaching computer programming”	00
SS06	CEIE	“habilidades de programação” OR “aprendizagem ativa de programação” OR “ensino de programação computadores”	29
SS07	ACM DL	“computer programming skills” OR “active programming learning” OR “teaching computer programming”	57
SS08	ACM DL	“habilidades de programacao” OR “aprendizagem ativa de programacao” OR “ensino de programacao computadores”	00
SS09	Engineering Village	“computer programming skills” OR “active programming learning” OR “teaching computer programming”	163
SS10	Engineering Village	“habilidades de programação” OR “aprendizagem ativa de programação” OR “ensino de programação computadores”	00
TOTAL			363

Tabela 1. Bases de pesquisa e *strings* de busca utilizadas

Os estudos encontrados nos resultados das buscas realizadas foram exportados em arquivos de referência no formato BIBTEX e catalogados utilizando uma ferramenta especializada para gestão de revisões sistemáticas, a ferramenta StArt desenvolvida pelo LaPES - Laboratório de Pesquisa em Engenharia de Software da UFSCar.

2.2. Critérios de inclusão e exclusão

A definição de critérios é de suma importância na pesquisa sistemática, uma vez que os motores de busca automatizada retornam muitos trabalhos, que em muitos casos não estão relacionados ao contexto da pesquisa, atenuando problemas na seleção de artigos por palavras-chaves com semântica distinta da estabelecida para investigação; o processo de busca sistemática retornou inicialmente 363 estudos, submetidos aos critérios de inclusão e exclusão dispostos na Tabela 2.

Critérios de Inclusão	Critérios de Exclusão
<p>CI01. Artigos publicados de janeiro/2014 a junho/2019; e</p> <p>CI02. Artigos publicados em português, inglês ou espanhol; e</p> <p>CI03. Artigos que propõem/relatam processos, ferramentas, ambientes, metodologias ativas de ensino ou estudos de caso envolvendo aprendizagem ativa de programação.</p>	<p>CE01 - Trabalhos não disponíveis para leitura completa (<i>download</i> ou <i>online</i>); ou</p> <p>CE02 - Artigos duplicados; ou</p> <p>CE03 - Estudos secundários ou segmentos de um mesmo projeto de pesquisa; ou</p> <p>CE04 - Estudos relacionados ao ensino de programação na Educação Infantil; ou</p> <p>CE05 - Estudos relacionados ao ensino de programação no Ensino Fundamental I; ou</p> <p>CE06 - Estudos relacionados ao ensino de programação no Ensino Fundamental II; ou</p> <p>CE07 - Capítulos de livros ou resumos de conferências.</p>

Tabela 2. Critérios de inclusão e exclusão aplicados aos estudos encontrados

2.3. Execução do protocolo

A seleção dos estudos primários foi realizada em quatro fases: pré-seleção, seleção inicial, seleção final e a extração e avaliação da qualidade dos estudos, ilustradas no infográfico da Figura 2.



Figura 2. Infográfico das etapas de seleção dos estudos candidatos

A fase de pré-seleção teve início com aplicação das *strings* de busca planejadas no protocolo da revisão, retornando 363 estudos das bases de pesquisa ilustradas na Figura 2, sendo estes armazenados na ferramenta StArt para depuração na fase seguinte.

A fase da seleção inicial dos estudos candidatos compreende a aplicação dos critérios de inclusão e exclusão estabelecidos no protocolo da revisão; analisando o título (*title*), resumo (*abstract*) e palavras-chave (*keywords*) dos trabalhos candidatos, são aceitos para a etapa seguinte os trabalhos que atendem todos os critérios de inclusão e não classificados em nenhum dos critérios de exclusão descritos na Tabela 2, apenas 26,44% dos trabalhos foram habilitados para fase seguinte, sendo identificados 41 trabalhos duplicados e rejeitados 196, estes últimos em muitos casos por terem sido publicados há mais de cinco anos ou não relatarem o aprendizado ativo de programação.

O estágio da seleção final de estudos candidatos é realizado por meio do *download* dos trabalhos selecionados na fase anterior, neste momento, a leitura dinâmica dos artigos utilizando a técnica *skimming* é realizada, sendo possível constatar com maior granularidade a observância dos critérios de inclusão ou algum critério de exclusão que não tenha sido identificado apenas com os descritores na fase anterior; nesta etapa 36,45% dos trabalhos foram rejeitados, principalmente por abordar o ensino de programação em séries do fundamental II, não disponíveis para *download*, estudos relacionados a capítulos de livro ou resumos de conferências, ou ainda trabalhos que não relataram em sua metodologia como o aprendizado ativo de programação foi mediado.

A extração dos dados e avaliação da qualidade dos estudos selecionados, compreende a leitura minuciosa e armazenamento de dados relevantes dos artigos identificados, descrita em detalhes nas seções seguintes.

2.4. Avaliação da qualidade dos estudos

A avaliação da qualidade dos estudos é imperiosa, sendo a qualidade uma preocupação transversal às etapas que disciplinam a revisão sistemática, permitindo desconsiderar estudos metodologicamente frágeis ou com resultados incipientes preponderantes para sua baixa qualidade científica em detrimento ao levantamento do estado da arte [Galvão and Pereira 2015]

As questões de qualidade formuladas, visam identificar a relevância do trabalho no que diz respeito ao objetivo principal estabelecido para pesquisa sistemática, bem como a confiabilidade da metodologia e resultados apresentados pelo estudo; os critérios adotados para avaliação da qualidade dos estudos foram:

- **QA01:** O estudo descreve o nível de ensino dos estudantes participantes?
- **QA02:** O estudo descreve qual metodologia ativa de ensino foi utilizada?
- **QA03:** Os resultados apresentados no experimento representam um contexto educacional real de ensino?

As questões foram ponderadas em conformidade ao seguinte barema: 1 (sim, o estudo selecionado satisfaz o critério); 0.5 (parcialmente, o estudo selecionado não relata claramente) e 0 (não, o estudo selecionado não satisfaz o critério), ao final da avaliação é realizado o somatório da pontuação obtida pelo estudo com os critérios de qualidade avaliados, àqueles com nota menor que 2.5 são descartados e eliminados da extração de dados [Kitchenham 2004].

2.5. Extração dos dados

A extração de dados compreende a etapa em que os dados dos estudos selecionados são extraídos e armazenados, a fim de que possam auxiliar na resposta às questões de pesquisa estabelecidas no protocolo; os atributos utilizados seguiram o template disposto parcialmente na Tabela 3.

Questões de Pesquisa	Atributo	Tipo de Dado
QP01 - Qual nível de ensino dos estudantes participantes do estudo?	Nível de Ensino	Lista pré-definida (Educação Infantil, Ensino Fundamental I, Ensino Fundamental II, Ensino Médio, Graduação, Pós-Graduação e Não Informado)
QP02 - Os estudantes são egressos de cursos não relacionados a computação?	Ensino de Programação para Estudantes de Computação?	Lista pré-definida (Sim, Não e Não Informado)
QP03 - Quais metodologias ativas de ensino apoiam o ensino de programação?	Qual Metodologia ativa de Ensino relatada pelo estudo no ensino ativo de programação?	Texto
QP04 - Quais ferramentas são utilizados para ensino de programação ?	Quais ferramentas utilizadas para ensino de programação?	Texto

Tabela 3. Template para extração dos dados

2.6. Estratégias para validação do estudo

A estratégia selecionada para a pesquisa sistemática envolve a participação de três pesquisadores, seguindo o proposto por Brasil, Medeiros and Nunes (2018) onde dois pesquisadores (R1 e R2) avaliam individualmente os estudos coletados na pré-seleção ilustrada na Figura 2, assinalando “aceito” ou “rejeitado” para cada estudo analisado; sendo que o trabalho ocorre de forma individual e independente em todas as etapas do processo de seleção dos estudos candidatos identificados, em caso de conflito no parecer quanto admissibilidade de determinado trabalho para fase seguinte, um terceiro avaliador externo (R3) é acionado emitindo um parecer “aceito” ou “rejeitado” para o estudo.

Ao final do processo, os estudos realizados por R1 e R2 são reunidos e tabulados em relatório final com informações dos estudos primários selecionados, a fim de que sejam aplicados os critérios de avaliação da qualidade, em seguida, os dados dos trabalhos selecionados são extraídos seguindo o template da Tabela 3, analisados e discutidos colaborativamente, culminando na divulgação dos resultados obtidos por meio da revisão sistemática da literatura.

3. Análise e Discussão dos Dados

A pesquisa sistemática identificou inicialmente, a partir das *strings* de busca constantes no protocolo aplicado a fase de pré-seleção dos trabalhos, um total de 363 estudos candidatos oriundos das mais diversas bases de pesquisa (*IEEE Xplore* 62; *ACM DLM* 57; *Scopus* 52; *Engineering*

Village 163 e o Portal de Publicações CEIE 29), sendo rejeitados 73,56% dos estudos por não atenderem concomitante os critérios de inclusão ilustrados na Figura 3a, ou ainda enquadrarem-se em um dos critérios de exclusão da Figura 3b.

Figura 3a. Seleção Inicial - Critérios de Inclusão

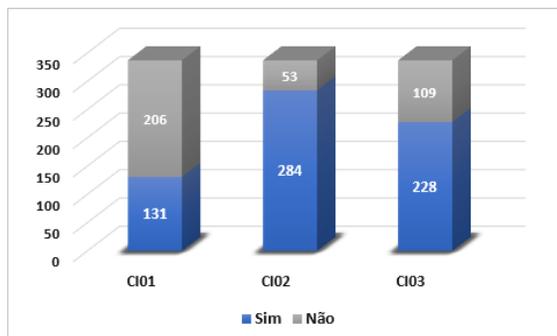
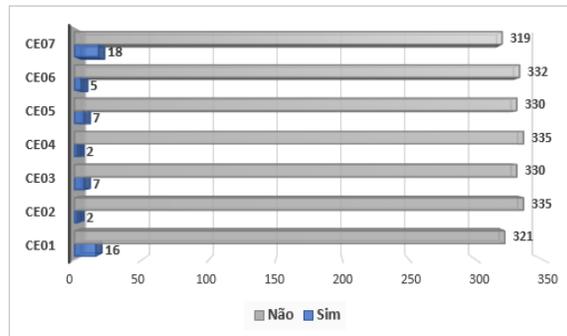


Figura 3b. Seleção Inicial - Critérios de Exclusão



Após realizada análise dos estudos, a partir da estratégia descrita na seção anterior, restaram vinte e nove trabalhos que foram alvo do processo de extração (ver Tabela 3) e avaliação de qualidade (ver Seção 2.4), gerando a Tabela 4 disponível em: <http://bit.ly/ppgite-dissertacao-rsl-papers>.

3.1. Respostas às questões de pesquisa

Os estudos selecionados na pesquisa sistemática e consolidados na Tabela 4, foram tabulados e indexados por um código identificador (ID), que será utilizado para referenciar determinado trabalho selecionado nas respostas às questões de pesquisa formuladas no protocolo, que em conjunto constituem o levantamento de estado da arte para a pesquisa.

QP01 - Qual nível de ensino dos estudantes participantes do estudo?

Em análise aos estudos selecionados, cerca de mais da metade dos trabalhos envolvem estudantes do ensino superior (58,62%), principalmente em disciplinas Introdutórias a Programação (ID: 64, 72, 1922, 2032, 2042 e 2135), ofertadas em cursos ligados à computação, engenharias diversas incluindo biomédica (ID 5), mecânica e elétrica (ID: 2036), elétrica (ID: 102) nos períodos iniciais da graduação, destaque para um experimento na universidade de Coimbra com estudantes de língua portuguesa (ID: 2086).

Um expressivo número de trabalhos têm sido desenvolvidos com alunos do ensino médio em disciplinas de informática (ID: 6), introdução à programação (ID: 20, 62, 90, 1887), educação em ciência da computação (ID: 7 e 1951), robótica (ID: 112) e interdisciplinar (ID: 24) ofertados na qualidade de componentes eletivos, enquanto no ensino técnico os estudos selecionados utilizam a robótica educacional como estratégia à promoção do aprendizado ativo de programação (ID: 1879, 1880 e 1882), objetivando desenvolver nos estudantes habilidades em pensamento computacional, raciocínio lógico e a resolução de problemas.

QP02 - Os estudantes são de cursos não relacionados a computação?

Apesar de grande parte dos estudos relacionarem estudantes da computação, cerca de 58%, observamos uma representatividade promissora no ensino médio (31,01%), bem como trabalhos interdisciplinares com alunos da computação e outros cursos (ID: 2036 e 2042).

QP03 - Quais metodologias ativas de ensino apoiam o ensino de programação?

O número expressivo de metodologias ativas de ensino identificados na pesquisa sistemática supra, superou às expectativas dos pesquisadores envolvidos, principalmente por algumas serem poucas conhecidas ou exploradas na literatura para ensino de programação, a exemplo da *challenge based learning* (ID: 1895), *inquiry-based learning* (ID: 72), *project-problem based learning* (ID: 22) e a *m-learning* (ID: 2135).

A *project-based learning* foi a metodologia ativa de ensino mais citada pelos estudos (ID: 5, 6, 7, 8, 12, 20, 24, 1951, 2036 e 2042), selecionada por proporcionar maior protagonismo, engajamento e motivação do estudante para aprendizado de programação através da construção de projetos, surgindo uma tendência envolvendo a incorporação de práticas em desenvolvimento ágil identificada em alguns estudos, seja com *scrum* (ID: 7 e 1895) ou *pair programming* (ID: 104).

QP04 - Quais ferramentas são utilizados para ensino de programação?

Não há um conjunto comum de ferramentas utilizadas nos estudos selecionados, exceto pelo *scratch* bastante popular no ensino de programação visual em blocos (ID: 6, 7, 24 e 1951) principalmente por seu aspecto lúdico e intuitivo, seguido pelo *hardware* arduíno comum em abordagens com robótica educacional ou pedagógica (ID: 5, 1880, 1882 e 2036), o Facebook citado em estudos envolvendo o ensino de programação com *blended learning* e *project-based learning* (ID: 1887 e 1895).

5. Considerações finais

A revisão sistemática da literatura realizada indica que poucas metodologias ativas de ensino são exploradas na construção de cenários educacionais envolvendo a aprendizagem ativa de programação, se limitando na maioria dos casos ao *Problem Based Learning* (PBL) ou *Project Based Learning* (PjBL), sendo uma lacuna a ser explorada em trabalhos futuros o impacto de outras metodologias ativas identificadas na aprendizagem ativa de programação.

Inúmeros estudos na área de computação e educação, apresentam ferramentas (ID: 1887; ID: 2036; ID: 2042) ou ambientes (ID: 64; ID: 90; ID: 112; ID: 2135) para ensino de programação; principalmente em cursos introdutórios ofertados para calouros (ID: 64; ID: 1882; ID: 1897; ID: 1922), contudo, não surtem o efeito esperado na experiência de aprendizagem destes estudantes, fato associado a inobservância na maioria dos cenários educacionais avaliados na RSL da metodologia ativa de ensino que guiará os processos de ensino e aprendizagem em conjunto com os objetos de aprendizagem propostos, sendo uma lacuna a ser explorada em trabalhos futuros.

Neste cenário, se faz necessário uma melhor difusão das metodologias ativas de ensino identificadas no levantamento de estado da arte realizado, incluindo construção colaborativa e compartilhando de planos de aula que envolvam boas práticas em sua utilização, atuando como estratégia fundamental a melhoria do fluxo da experiência de aprendizagem, ressignificação do papel na mediação docente-discente e conseqüente melhoria dos resultados educacionais concernentes às disciplinas ou cursos de programação; dentre as metodologias identificadas destacamos: *blended learning* (b-learning), *game-based learning* (GBL), *gamification*, *inquiry-*

based learning (IBL), m-learning, pair programming, problem-based learning (PBL), problem-project based learning (PPBL), project-based learning (PBL or PjBL), challenge based learning (CBL), robótica educacional e pedagógica.

Referências

- Brasil, Pedrina, Medeiros, Tainá Jesus and Nunes, Isabel Dillmann. Uma Revisão Sistemática sobre o uso de Learning Analytics em ambientes virtuais de aprendizagem brasileiros, In: Ctrl+E Congresso sobre Tecnologias Educacionais, Fortaleza - CE, 2018.
- Galvão, Taís Freire and Pereira, Mauricio Gomes (2015). Avaliação da qualidade da evidência de revisões sistemáticas. *Epidemiologia e Serviços de Saúde*, v. 24, p. 173-175.
- Kitchenham, Barbara (2004). *Procedures for performing systematic reviews*. Keele, UK, Keele University, v. 33, n. 2004, p. 1-26.
- Moran, José (2018). Metodologias ativas para uma aprendizagem mais profunda. Metodologias ativas para uma educação inovadora: uma abordagem teórico-prática. Porto Alegre: Penso, p. 02-25.
- Silva, Thiago Reis da et al. (2015). Ensino-aprendizagem de programação: uma revisão sistemática da literatura. *Revista Brasileira de Informática na Educação*, v. 23, n. 1.
- Stefanova Ratcheva, Vesselina and Till Leopold (2018). *5 things to know about the future of jobs*, World Economic Forum.