

Uso de Rubricas em Pesquisas de Informática e Educação

- Uma Revisão Sistemática da Literatura

Dirson Santos de Campos
Instituto de Informática
Universidade Federal de Goiás (UFG)
Goiânia-GO, Brasil
dirson@inf.ufg.br

Deller James Ferreira
Instituto de Informática
Universidade Federal de Goiás (UFG)
Goiânia-GO, Brasil
deller@inf.ufg.br

RESUMO

Este artigo apresenta os resultados de uma Revisão Sistemática da Literatura (RSL) que incluiu a análise de artigos sobre rubricas usadas no processo de aprendizagem de programação em três importantes bases internacionais, nomeadamente ScienceDirect, Scopus e Web of Science. A ferramenta Parsifal foi usada para apoiar a execução da RSL. A rubrica usada em pesquisas de aprendizagem fornece uma ampla gama de critérios de avaliação e padrões de desempenho esperados para o desenvolvimento educacional e de habilidades, proporcionando aos alunos atividades vinculadas aos objetivos pedagógicos e de aprendizagem. O conteúdo da rubrica, em várias dimensões, produz métricas robustas para avaliar a aprendizagem do aluno. Apesar da rubrica ser amplamente utilizada na educação, a sumarização da literatura acadêmica sobre este tema na aprendizagem de programação é extremamente rara. Aplicando o protocolo de pesquisa da RSL aproximadamente 200 artigos inscritos para seleção do estudo. O título, resumo e palavras-chave foram analisados para encontrar os estudos que atenderam aos critérios de inclusão. Em seguida, foram aplicados os critérios de exclusão, cerca de 11% foram aceitos. Foi feito uma leitura completa dos artigos aceitos. Nossa categorização de recursos foi usada para auxiliar a extração de dados do Parsifal. A categorização aumenta a qualidade dos dados exportados do Parsifal para Excel. Os resultados obtidos fornecem uma compreensão de como rubricas são exploradas no aprendizado de programação, recursos usados para aplicá-los e desafios neste processo, especialmente relacionados ao estudo empírico, *feedback* e a definição de rubricas aplicadas à aprendizagem de programação.

Fica permitido ao(s) autor(es) ou a terceiros a reprodução ou distribuição, em parte ou no todo, do material extraído dessa obra, de forma verbatim, adaptada ou remixada, bem como a criação ou produção a partir do conteúdo dessa obra, para fins não comerciais, desde que sejam atribuídos os devidos créditos à criação original, sob os termos da licença CC BY-NC 4.0.

EduComp'21, Abril 27–30, 2021, Jataí, Goiás, Brasil (On-line)

©2021 Copyright mantido pelo(s) autor(es). Direitos de publicação licenciados à Sociedade Brasileira de Computação (SBC).

PALAVRAS-CHAVE

Revisão Sistemática da Literatura (RSL), Rubricas, Aprendizagem de Programação, Avaliação, *Feedback*.

1 Introdução

Mensurar a aprendizagem dos estudantes de forma sistemática e justa é uma atividade complexa, seja pela dificuldade de compreensão dos objetivos e critérios de avaliação pelo discente, seja pelo forte caráter subjetivo intrínseco ao processo avaliativo docente. Em disciplinas de programação um mesmo erro lógico detectado pode ter *feedbacks* não padronizados e resultados avaliativos dispares, de acordo com o rigor aplicado à avaliação de um erro específico, pois existem critérios implícitos e explícitos aplicados na avaliação pelos docente de acordo com objetivos didáticos definido por ele mesmo a tarefa. O mesmo ocorre quando se fornece *feedback* aos alunos que pode ter conteúdos de diferentes níveis, desde os triviais aos mais pedagogicamente elaborados.

Uma das formas eficientes de padronizar e instrumentar aspectos relativos à avaliação discente é a adoção de rubricas. A rubrica aplicada a educação é um tipo de instrumento pedagógico que obriga a explicitar os critérios de avaliação e minimizar os efeitos de critérios difusos nesta mesma avaliação. As rubricas podem ser usadas para avaliar uma grande variedade de atribuições e tarefas discentes ou de pesquisa [1]: exercícios, tarefas, seminários, artigos de pesquisa, críticas de livros, participação em fóruns, relatórios de laboratório, portfólios, tarefas em grupo, apresentações orais, questionários discursivos entre outros.

Em relação a avaliação de pesquisa acadêmicas Vaccari e Thangam [29] criaram e testaram em várias defesas de doutorado uma rubrica que se mostrou adequada para avaliar os estudantes de programas de pós-graduação em ciência e engenharia. A rubrica proposta pelos autores detalhava os critérios didáticos da avaliação da dissertação de doutorado e também da defesa oral o que comprova a versatilidade de adaptação da rubrica para diversos níveis de avaliação discente.

Em Estudos Empíricos a rubrica, em suas diversas dimensões, pode auxiliar tanto na definição padronizada dos dados que serão coletados para avaliação discente quanto na estratégia de análises dos mesmos de modo que sejam consistentes e abrangentes com objetivos didáticos contemplados pela rubrica.

Em sala de aula a rubrica tanto pode ser usada para avaliação de estudantes quanto para análise e mensuração do processo de ensino-aprendizagem. No primeiro caso, quaisquer artefatos produzidos pelos estudantes, ao longo da disciplina, podem ser avaliados e as rubricas utilizadas para aferição [2]. No segundo caso, a rubrica explicita, à priori, os critérios bem definidos e delineados da avaliação e os seus níveis de desempenho desejáveis de acordo com os objetivos de aprendizagem da disciplina e pode servir como guia do processo de aprendizagem e como parâmetro no desenvolvimento, revisão e julgamento deste processo [3].

A rubrica é amplamente utilizada na educação superior e em estudos empíricos em sala de aula [3, 5, 6, 7, 8, 9, 10]. Além disso, possui suporte de Ambientes Virtuais de Aprendizagem (AVA) importantes, tais como, Moodle, Google Classroom, Blackboard AVA entre outros. Possui suporte de ferramentas especializadas específicas, por exemplo, a RubStar [4], mantida pela Universidade do Kansas (USA) que possui mais de 18000 modelos de rubricas informatizadas cadastradas. Apesar disso, a sumarização da literatura acadêmica sobre este tema na aprendizagem de programação é extremamente rara e uma Revisão Sistemática da Literatura (RSL) envolvendo rubricas e a aprendizagem de programação é relevante devido a importância do tema.

Foi utilizada a ferramenta *on-line* Parsifal (<https://parsifal.com>) para definir, planejar e executar a RSL. Esta ferramenta foi projetada para atender os passos do modelo de revisão sistemática proposto por Kitchenham e Charters [11].

A primeira contribuição deste artigo é a própria RSL sobre o tema em três importantes bases, a saber, *ScienceDirect*, *Scopus e Web of Science*. As três bases foram escolhidas pela sua abrangência e relevância na literatura científica na língua inglesa e estarem disponíveis no portal de periódicos da CAPES. Outras contribuições secundárias, mais significativas no contexto da pesquisa deste artigo são:

- Uma nova categorização de rubricas baseadas na análise sistematizada dos artigos selecionados na RSL mais apropriada ao ensino e aprendizagem de programação que tem características e ferramentas distintas de outras disciplinas. Estas categorias se encontram na Tabela 4.
- Dicas práticas para auxiliar a manipulação dos dados exportados pela ferramenta Parsifal de modo a criar novos gráficos não suportados pelo módulo de análise de dados desta ferramenta. Pode ser útil a outros pesquisadores que pretende fazer uma RSL pela ferramenta ou estão em doutoramento pois a RSL pode ajudar a encontrar gaps em pesquisas.

2 Trabalhos Relacionados

A rubrica pode ser usada como um método de pesquisa à medida que as rubricas são adotadas e adaptadas em diferentes contextos de pesquisas. Entretanto relatos detalhados de adaptações de rubricas à pesquisa que descrevem como foi mapeada os objetivos pedagógicos da avaliação e descrever o quão bem a rubrica mensura a aprendizagem dos estudantes são relatos um tanto raros [30].

Os trabalhos relacionados nos ajudam a compreender como outros pesquisadores usam rubricas implícitas ou explícitas envolvendo disciplinas e fundamentos da programação mapeados para os objetivos inerente a pesquisa deles.

Amal et. al. [5] propõe uma avaliação formativa baseada em tarefas para programadores novatos em cursos de programação por meio da comparação dos resultados de uma rubrica e um formulário de avaliação da qualidade da programação com cinco categorias ou dimensões da rubrica para mensurar o trabalho discente: correção, estrutura, clareza, *layout* e exceções. Em seguida, foi medida a eficácia do método mensurando as pontuações dos alunos em um exame de meio do semestre. Os resultados da avaliação estatística sugeriram que a avaliação dos pares afetou positivamente a competência de programação dos estudantes, porém os autores ressaltam que o resultado não é definitivo. Segundo os autores para generalizar os resultados obtidos no estudo empírico executados por eles em disciplinas do tipo CS1 (Curso introdutórios de Computação da *ACM Computing Curricula*) a abordagem empírica deve ser aplicada a alunos que estudam em diferentes níveis de programação, a fim de determinar mais claramente como a experiência de programação pode afetar os resultados da avaliação por pares.

Zhang et. al. [6] usa um projeto OO (Orientado a Objetos) chamado “Funcionário do Departamento” como pano de fundo do estudo empírico executado por eles. A rubrica é usada em sua maneira clássica, como um instrumento de avaliação das atividades do projeto padronizada, desde sua especificação até avaliação e através dela, gerando relatórios de pontuação. Depois de receber as submissões dos projetos dos estudantes, o professor irá avaliar e atribuir uma pontuação a cada uma das submissões de acordo com uma rubrica de avaliação predefinida. Tanto a nota do projeto quanto a nota final do curso serão coletadas ao final de cada semestre. O coeficiente de correlação será calculado e um gráfico de dispersão será criado para mostrar a relação entre eles. As percepções dos estudantes serão coletadas em um questionário *online* feito usando a ferramenta *online* chamada *Survey Monkey* (<https://www.surveymonkey.com/>). O questionário inclui nove perguntas: as duas primeiras foram questões dissertativas, que estão relacionadas às percepções dos estudantes sobre o projeto as demais foram de múltiplas escolhas.

Dorodchi et. al. [7] executaram um estudo empírico em sala de aula em disciplinas introdutórias de programação (CS1) onde a rubrica pedagógica foi desenvolvida para classificar questões neste contexto, no caso para descrever os domínios cognitivos da Taxonomia de Bloom de forma a tentar classificar o estudante como no meta-nível cognitivo de iniciante, intermediário ou avançado. A rubrica desenvolvida foi aplicada para avaliar três testes semestrais e para auxiliar a elaborar um exame final. Na avaliação final, os pesquisadores propuseram questões para desafiar as habilidades dos estudantes em uma proporção e combinação predeterminadas que mapeiam os níveis da Taxonomia de Bloom para a rubrica. Depois de cada avaliação, as dificuldades dos estudantes foram identificadas e as atividades propostas aos discentes durante o período de aula foram ajustadas para resolver as deficiências discentes detectadas.

Garces et. al. [8] propuseram uma rubrica para avaliação holística do desempenho de novatos em disciplinas introdutórias de programação analisando 11 tarefas submetidas em linguagem C. Não foi usada uma ferramenta de correção automática baseada em testes automatizados detectando erros lógicos de forma direta e sim foi analisando os comentários dos códigos fontes para determinar como os estudantes compreenderam ou explicaram os exercícios. A análise qualitativa era baseada no conteúdo da rubrica em quatro níveis de complexidade da mais básico até o mais completo.

Ahoniemi et. al. [9] usa a rubrica de avaliação para padronizar e melhorar o *feedback* pedagógico apresentado aos estudantes novatos e fornecer uma avaliação objetiva dos erros discentes de acordo com critérios explícitos de qualidade descritos na rubrica. Não há correção automática de exercícios com ferramentas, porém a rubrica ajuda a padronizar o conteúdo escrito do *feedback* usando uma ferramenta interna, denominada ALOHA, desenvolvida pelo próprio grupo de pesquisa dos autores. Apesar de existirem muitas ferramentas para oferecer suporte à avaliação totalmente automática, como ferramentas online Judge como o BOCA por exemplo, a ferramenta dos autores apresentada possui um diferencial interessante porque possui recursos para suportar uma avaliação manual e em especialmente, o auxílio para a definição de *feedback* por escrito.

Salankar [10] analisa três Estudos de Caso cujo o objetivo geral é identificar o impacto de duas rubricas diferentes. A primeira rubrica com sete critérios ou dimensões e um nível de avaliação, a segunda com quatro critérios e quatro níveis de satisfação. As rubricas foram construídas para o modelo educacional específico chamado ADDIE [12]. ADDIE foi criado para cursos *online* onde os critérios de avaliação dos estudantes dever ser claros e colocados à priori. O autor usou o ADDIE objetivando clarificar critérios de avaliação, no caso, para avaliar o desempenho dos alunos em disciplina do segundo curso de programação CS2 (*CS2 of ACM Computing Curricula*). O resultado foi mensurado em dois grupos um antes e um depois do uso da rubrica e também foi analisado as diferenças de gênero dos estudantes. Os resultados do uso da rubrica foram estatisticamente superiores comprovado pelo teste t de *Student* que é um teste de hipótese que usa conceitos estatísticos para rejeitar ou não uma hipótese nula quando a estatística de teste (t) segue uma distribuição t de *Student*.

3 Metodologia

Este artigo apresenta os resultados de uma Revisão Sistemática da Literatura (RSL) que incluiu pesquisas sobre rubricas no processo de aprendizagem de programação em três importantes bases internacionais disponíveis nos periódicos da Capes, nomeadamente ScienceDirect, Scopus e Web of Science.

A ferramenta Parsifal foi usada para apoiar a RSL. Parsifal é uma ferramenta robusta construída com base nos trabalho proposto por Kitchenham e Charters [11] que possui etapas que permitem identificar, planejar, conduzir, extrair dados, avaliar e interpretar estudos primários ou mesmo secundários disponíveis em bases científicas relevantes para responder questões de pesquisa sobre um determinado tópico ou área temática de pesquisas.

3.1 Questões de Pesquisa

Para delinear o escopo e direcionar o desenvolvimento desta RSL, as questões de pesquisa foram elaboradas com base na estratégia PICOC (*Population, Intervention, Comparison, Outcome e Context*) adotada pela ferramenta Parsifal. Seguindo essa estratégia, a RSL tem como questões de pesquisas:

- QP1 – “Como as rubricas são usadas em pesquisas empíricas e científicas na aprendizagem de programação no ensino superior?”
- QP2 – “Que tipo de rubricas são usadas na aprendizagem de programação no ensino superior?”

3.2 Planejamento da RSL

Foi realizada uma busca por documentos científicos disponibilizadas pelo Portal de Periódicos CAPES, à saber, *journals*, anais de eventos *online*, capítulos de livros em três importantes e relevantes bases internacionais *ScienceDirect*, *Scopus e Web of Science*, sendo estas consideradas fontes de pesquisa científica internacionais que possuem um amplo conjunto de publicações relacionadas à Informática e Educação.

3.2.1. Expressão de Busca

A expressão de busca nos permite analisar os resultados obtidos a cada execução da *string* nas bases pesquisadas. Em nossa RSL, visando que os resultados estejam alinhados aos objetivos da pesquisa selecionando documentos onde a rubrica seja relevante a pesquisa, a *string* de busca deve pesquisar os termos no título, resumo ou palavras-chave. É importante citar que as opções de filtragem fornecidas pelas bases também integraram o processo da busca, foram selecionados, por exemplo, o idioma de escrita dos artigos, no caso o inglês permite uma pesquisa mundial.

Para a elaboração da expressão de busca, utilizou-se palavras-chave que representam no PIPOC por *Population, Intervention e Context*. A expressão de busca no Parsifal da nossa pesquisa é:

(("programming" AND "learning") AND ("rubrics" OR "rubric"))

Pequenas variações na sintaxe da expressão de busca são necessárias sempre que os parâmetros do mecanismo de busca das bases utilizadas exigirem.

3.2.2 Critérios de Inclusão e Exclusão

Os critérios de inclusão utilizados foram: avaliação ou verificação, *feedback*, metodologias de pesquisa em educação aplicadas a Ciências da Computação ou Engenharia, pesquisas em artigos de conferências, *journals* e capítulos de livros em inglês, estudos que relatem abordagens de mensuração de objetivos de aprendizagem com o auxílio de rubricas.

Os critérios de exclusão foram: *poster, short papers*, documentos repetidos ou redundantes (artigos publicados pelo mesmo autores em veículos diferentes com variações de conteúdo não significativa), documentos pago ou que não estejam disponíveis para leitura pela plataforma CAPES, documento em idioma diferente do inglês, estudos empíricos baseado em games, baseado em robótica, estudos que envolvam disciplinas da Ciência da Computação ou da Engenharia mas que não envolvam programação, estudos que focam em outros assuntos educacionais que não seja rubricas, estudos educacionais que envolvam rubricas mais aplicados a outras áreas como saúde e matemática, estudos envolvendo aprendizagem de programação em níveis de estudos elementares ou secundários, em inglês, usa-se os termos K-6 e K-12 indicando os anos de estudos da criança ou adolescente em alguns países, por exemplo, os Estados Unidos da América.

4 Condução da RSL

A busca nas três bases internacionais já citadas e a seleção dos documentos ocorreram de acordo com os critérios de inclusão e exclusão, bem como a avaliação de qualidade, ocorreram entre maio de 2020 a setembro de 2020.

O processo de seleção dos trabalhos foi dividido em cinco fases de acordo com as opções da ferramenta Parsifal, a saber:

- Fase de Busca (opção *Search* do Parsifal) nas três bases internacionais *ScienceDirect*, *Scopus* e *Web of Science*,
- Fase de Importação de Estudos (opção *Import Selection*),
- Fase de Seleção de Estudos (opção *Study Selection*),
- Fase de Extração de Dados (opção *Data Extraction*) e
- Fase de Análise dos Dados (opção *Data Analysis*).

Em cada uma destas fases foram feitas o tipo de verificação apropriada a fase e análise correspondente que foi aplicada em cada artigo.

4.1 Fase Busca (*Search*)

A fase de busca, auxiliada pelo Parsifal utiliza-se das funcionalidades da opção *Search*. O Parsifal está integrado com as bases *Scopus* e *ScienceDirect*, mas não como a *Web of Science*.

As bases integradas a ferramentas possuem funcionalidades que facilitam o processo de importação, tais como, busca na base diretamente do Parsifal e salvar a expressão ou *string* de busca com sintaxe específica da base *Scopus* e *ScienceDirect*.

A base *Web of Science* não está integrada e a interface de busca da ferramenta é feita diretamente no mecanismo de busca da referida base. É importante estar logado no Portal Periódicos da CAPES na comunidade acadêmica federada (CAFE) para ter acesso aos documentos.

4.2 Fase Importação dos Estudos (*Import Selection*)

A Fase de Importação dos Estudos (*Study Selection*) recebe com *input* os estudos selecionados na seção 4.1.

A Figura 1 ilustra os estudos importados de cada uma destas das bases internacionais.

Import Studies	
Source	Imported Studies
Science@Direct	68 
Scopus	88 
Web of Science	40 

Figura 1: Estudos importados das três bases da RSL

Analisando a Figura 1 verifica-se a importação de 196 documento ao aplicar a *string* de busca da seção 3.2.1. Observe que na Figura 1 é disponibilizada um botão de importação para cada base. A importação foi feita em arquivos no formato BibTeX (.bib) para cada uma das bases. Para gerar os arquivos do tipo .bib é necessário selecionar todos os documentos retornados ao executar a *string* de busca em cada uma das três base. Isto é feito usando o botão tipo *CheckBox* da interface de consulta da base no Parsifal.

4.3 Fase Seleção de Estudos (*Study Selection*)

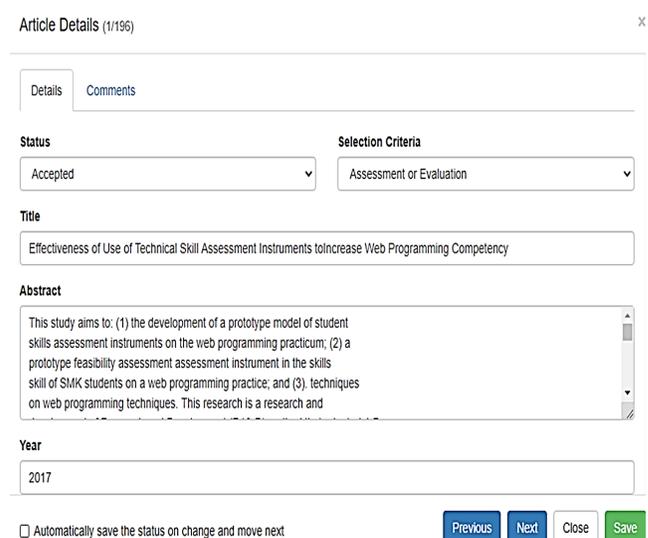
A Fase de Seleção de Estudos é feita com base em metadados que são preenchidos pelos pesquisadores e armazenados na ferramenta Parsifal conforme foi ilustrado na seleção do artigo da Figura 2.

Conforme visto na Figura 2 na aba *Details* pode-se ver alguns dos metadados importados no formato .bib e usados como parâmetros da seleção. Nos utilizamos os parâmetros títulos, resumo e palavra-chave (*keyword* não visualizado na figura 2 em razão das restrições do tamanho da figura).

Para cada um dos 196 artigo estas metadados são preenchidos conforme os critérios de inclusão para artigo com *status* aceitos (*Accepts*). Na figura 2 o critério de inclusão principal explica o porquê a artigo obteve o status aceito (*Accept*) é usar a rubrica para avaliação, em inglês, *Assessment or Evaluation*. Depois de preencher os metadados é necessário salvar as informações que ficam, assim, disponibilizadas ao pesquisador na ferramenta.

Analogamente os critérios de exclusão são utilizados para eliminar aqueles documentos que não têm como foco nos objetivos desta RSL, ou seja, estudos envolvendo rubricas fora do contexto da aprendizagem de programação ou dos seus fundamentos.

No Parsifal os resultados possíveis para a seleção de um artigo são: aceitos, rejeitados, não classificados e duplicados. A Figura 2 exemplifica os metadados que são disponibilizados de um dos artigos [13] selecionados para a RSL.



The screenshot shows the 'Article Details' window for article 11/196. It features a 'Details' tab and a 'Comments' tab. The 'Status' is set to 'Accepted' and the 'Selection Criteria' is 'Assessment or Evaluation'. The title is 'Effectiveness of Use of Technical Skill Assessment Instruments to Increase Web Programming Competency'. The abstract describes a study on developing a prototype model for student skills assessment instruments on web programming practice. The year is 2017. At the bottom, there are buttons for 'Previous', 'Next', 'Close', and 'Save', along with a checkbox for 'Automatically save the status on change and move next'.

Figura 2: Exemplo de Seleção de Estudos usando os metadados disponibilizados na Ferramenta Parsifal

O resultado final desta análise permite a classificação do *status* do documento em relação a RSL e esses resultados de seleção dos 196 documentos usando critérios de inclusão e exclusão como ilustrado na Figura 2 foram sintetizados na Tabela 1.

Tabela 1: Questões, tipos de dados e valores típicos usados para extração de dados do Data Extraction Form do Parsifal

Tipo de Resultado	Quantidade de artigos deste resultado (porcentagem)
Aceitos	21 (10,72%)
Rejeitados	121 (61,73%)
Não Classificados	0 (0%)
Duplicados	54 (27,55%)
Total	196 (100%)

4.4 Fase Extração de Dados (Data Extraction)

A Fase de Extração de dados da RSL foram apoiadas pelo Parsifal que auxilia a definição de questões do Formulário de Extração de Pesquisa (Data Extraction Form) da ferramenta.

As questões incluídas no formulário nos permite extrair informações do *data set* sobre o QP1 e QP2 (seção 3.1). Essas questões estão descritas na Tabela 2 que incluem:

- uma descrição das questões;
- o tipo de dados que será retornado e os
- valores possíveis deste retorno quando aplicado.

Tabela 2: Questões, tipos de dados e valores típicos usados para extração de dados do Data Extraction Form do Parsifal

Description	Type	Values
Apresenta um estudo empírico em sala de aula?	Boolean Field	n/a
Usa Linguagem de Programação em que tipo de disciplina?	Select One Field	<ul style="list-style-type: none"> • CS0 • CS1 • CS2 • It does not involve programming disciplines • Multiple levels of programming discipline • Other programming discipline
Houve algum tipo de feedback durante o período de aula da disciplina?	Select One Field	<ul style="list-style-type: none"> • Não • Não aplicado, pois não envolve atividade em sala de aula • Sim
Houve algum tipo de correção automática ou semi-automática de exercícios discentes?	Select One Field	<ul style="list-style-type: none"> • Não • Não se aplica pois não envolve correção de exercícios • Sim
Qual o tipo de rubrica foi usada pelo pesquisador?	Select One Field	<ul style="list-style-type: none"> • Adaptive rubrics. • Assessment rubrics. • Formative rubrics. • Multiple rubrics. • Pedagogical rubrics. • Rubrics for Research Methods. • Standardized rubrics.

Observe que os dados coletados descritos na Tabela 2 nos permite definir dois tipos de categorizações empíricas da nossa RSL:

- O primeiro tipo de categorização é sobre linguagens de programação utilizados pelos pesquisadores em seus trabalhos. Refere-se, principalmente, a questão de pesquisa QP1.

- O segundo tipo de categorização é sobre o tipo de rubrica que na prática é utilizada pelos pesquisadores em seus trabalhos. Refere-se, principalmente, a questão de pesquisa QP2.

As demais questões da Tabela 2 nos ajudam a analisar os resultados da RSL e também subsidia na discussão dos resultados e conclusões obtidas.

Para cada um dos 21 artigos aceitos na RSL uma leitura completa foi feita e para cada um deles foi respondidas as questões da Tabela 2 conforme pode ser visto na Figura 3 que nos permite visualizar um exemplo do tipo de informações disponibilizadas pela ferramenta Parsifal em um dos artigos selecionados como aceitos.

Na Figura 3 o nome do artigo aparece na primeira linha e as respostas são obtidas de acordo com o tipo (*Type*) e valores (*Values*) definido da Tabela 2.

Evaluating a peer assessment approach in introductory programming courses

Apresenta um estudo empírico em sala de aula?

True

Usa Linguagem de Programação em que tipo de disciplina?

CS1

Houve algum tipo de feedback durante o período de aula da disciplina?

Sim

Houve algum tipo de correção automática ou semi-automática de exercícios discentes?

Não

Qual o tipo de rubrica foi usada pelo pesquisador?

Assessment rubrics.

Figura 3: Exemplo de análises de um artigo com os resultados obtidos ao responder as questões da Tabela 2

4.5 Fase Análise de Dados (Data Analysis)

Os resultados referentes a seleção dos artigos usando critérios de inclusão e exclusões foram colocados na Tabela 1.

Os resultados da Tabela 2 serão descritos na seção 5 deste artigo pois serão utilizados para responder as questões de pesquisa QP1 e QP2 (seção 3.1).

5 Análise dos Resultados da RSL

Após a aplicação dos critérios de inclusões e exclusões foram selecionados 21 artigos. Os resultados apresentados na Tabela 1 podem ser manipulados pela própria ferramenta que tem um módulo de Análise de dados (*Data Analysis*) que possui três tipos de gráficos pré-definidos: artigos por bases, artigos aceitos por base e artigos aceitos por ano, disponíveis após a seleção dos estudos e avaliação da qualidade pelos revisores.

A Figura 4 nos mostra um panorama a respeito dos artigos aceitos em nossa RSL. Observa-se que os números de artigos envolvendo rubricas e aprendizagem de programação vem crescendo entre 2019 e 2020, embora não seja muitos que envolvem rubricas e programação.

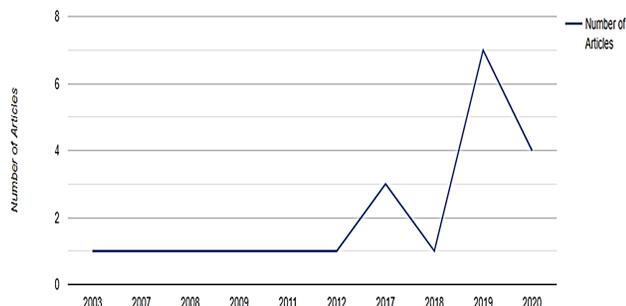


Figura 4: Números de artigos publicados por ano nas bases de pesquisas após a seleção do estudo e avaliação da qualidade feita pelos revisores desta RSL.

5.1 Resposta a QP1

A seção 2 deste *paper* também auxiliam a responder a QP1, mas por razões de melhor fluidez do relato desta pesquisa serão condensadas e discutidas na seção 7 porque a análise dos trabalhos relacionados não vem diretamente somente das questões que nortearam a extração de dados da Tabela 2 do *Data Extraction Form* da ferramenta Parsifal e sim de uma avaliação holística dos revisores baseado nas leituras de cada artigo selecionado.

Nesta seção discutiremos a resposta QP1 com base dos resultados obtidos pela análise dirigida pelas funcionalidades da ferramenta Parsifal. No intuito de responder como as rubricas são usadas em pesquisas empíricas e científicas na aprendizagem de programação no ensino superior (QP1) foi investigado:

- Se a rubrica foi usada em um estudo empírico em sala de aula.
- Qual ou quais as linguagens de programação foram utilizadas na disciplina pelo pesquisador.
- Se houve algum tipo de *feedback* durante a disciplina ou a análise dos pesquisadores foi feita após o encerramento da disciplina.

A rubrica foi usada em um estudo empírico em sala de aula em 85,72% das pesquisas [5, 6, 7, 8, 9, 10, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 27] e em 14,28% não foi usada [24, 25, 26].

Com relação a linguagem de programação foi utilizada na disciplina pelo pesquisador os dados extraídos do Parsifal nos permite montar uma categoria importante uma vez que a escolha da linguagem e de seu paradigma influencia sobremaneira tanto as estratégias de ensino adotada pelo docente quanto de aprendizagem pelo discente.

A Tabela 3 nos mostra a categorização obtida de forma empírica usadas nos artigos aceitos usando a opção exportação de dados do *Data Extraction* do Parsifal.

Na Tabela 3 a linguagem de programação e descrita pelos revisores ou é utilizado a classificação internacional da ACM, usa-se para disciplinas introdutórias o termo usado é CS1 e CS2 para a segunda disciplina de programação que é associada ao paradigma

da orientação a objetos. Outros três aspectos foram também considerados a respeito das linguagens que são:

- Se não envolvem conteúdo de uma disciplina de programação específica e sim fundamentos da lógica de programação.
- Se a pesquisa se refere a vários paradigmas de linguagem de programação usadas no ensino de programação.
- Se uma sintaxe específica de uma linguagem de programação de um módulo de uma ferramenta, por exemplo, comando da linguagem *shell script* de uma ferramenta voltada ao ensino do Sistema Operacional.

Foi analisado também se houve algum tipo de *feedback* durante a disciplina. Em 66,66% das pesquisas foi dado algum tipo de *feedback* durante a disciplina, isto ocorreu nos artigos [5, 7, 8, 9, 10, 14, 15, 16, 17, 18, 19, 20, 24, 27]. Em 33,34% não houve qualquer tipo de *feedback* descrito pelos autores dos artigos, isto ocorreu em [6, 13, 21, 25, 22, 23, 26].

Tabela 3: Categorização das linguagens de programação usadas pelos pesquisadores obtidas empiricamente na RSL

Descrição do Tipo ou classificação internacional da Linguagem de Programação	Porcentagem deste resultado [artigo analisado]
CS1	23,81 % [5, 7, 8, 9, 15]
CS2	19,05% [6, 10, 25, 27]
Não envolve disciplinas de programação específica e sim fundamentos da lógica de programação	14,28% [18, 20, 24]
Vários paradigmas de disciplina de programação	26,08% [16, 17, 19, 22, 23, 26]
Programação, com sintaxe específicas, de um módulo de uma ferramenta.	14,28% [13, 14, 21]
Total	100%

Foi analisado se houve algum tipo de correção automática ou semi-automática de exercícios discentes nos estudos pesquisados, as respostas foram:

- Em 66,67%, não houve correção automática ou semi-automática dos códigos fontes. Os pesquisadores que fizeram esta prática são [5, 6, 8, 10, 13, 14, 17, 18, 19, 21, 22, 23, 26, 27].
- Em 19,05%, houve correção automática ou semi-automática dos códigos fontes. Os pesquisadores que fizeram optaram por esta solução são [5, 9, 15, 16, 25].
- Em 14,28% não se aplica, pois não envolve correção de exercícios durante a pesquisa, somente questionários qualitativos sobre temas referentes a aprendizagem, isto ocorreu em [7, 20, 24].

5.2 Resposta a QP2

A seção 2 também auxiliam a responder a QP2 e a sua análise foi condensada e discutida na seção 7 porque também não é baseada na Tabela 2 do *Data Extraction Form* da ferramenta Parsifal.

Nesta seção discutiremos a resposta QP2 com base dos resultados obtidos pela análise dirigida pelas funcionalidades da ferramenta Parsifal.

Existem categorização clássicas de rubricas relatada em pesquisas da Ciência da Educação [1, 3, 10], mas outras não são categorizadas como as clássicas e são definidas pelas necessidades dos autores de forma experimental [2, 4, 14, 16, 18].

As rubricas são de grande importância na Ciências da Educação bem como na Informática na Educação. As razões já foram descritas por diversos autores [1, 28], as principais são:

- As rubricas facilitam a comunicação dos objetivos esperados e padrões de qualidade de tarefas a outras pessoas no ambiente acadêmico ou profissional.
- As rubricas em ambiente acadêmico ajudam a refinar os métodos de ensino adotados pelo *staff* da disciplina.
- As rubricas nivelam o ambiente de avaliação explicitando critérios implícitos e explícitos.
- Rubricas podem ser usadas em para acompanhar o desenvolvimento de habilidades discentes, definir objetivos de aprendizagem e critérios de avaliação em evolução desta mesma aprendizagem.
- Rubricas podem auxiliar o planejamento de tarefas, fazer avaliações formativas, sumativas centrada no desempenho, domínio, aprendizagem autodirigida e avaliação.

Outras características das rubricas:

- As rubricas necessitam serem feitas sob medida para as tarefas ou produtos que se pretende avaliar;
- Rubricas precisam descrever níveis de desempenho, de competência, na realização de tarefas específicas, ou de um produto específico.
- No seu conjunto, esses níveis de competência, descrevem qualquer resultado possível sobre o desempenho de um aluno;
- Rubricas determinam expectativas de desempenho para os discentes.

É importante citar, no contexto deste artigo, que em estudos empíricos a rubrica auxilia a padronizar o próprio estudo de modo que a coleta e análise de dados sejam consistentes e abrangentes, sendo assim, auxilia a supra análise padronizada dos dados coletados em um domínio específico da pesquisa.

Neste contexto uma rubrica aplicada a uma disciplina ou trabalho acadêmico é uma das soluções para se obter a objetividade e a consistência do julgamento do aluno, em um trabalho acadêmico mais complexo, como um seminário manter a objetividade e consistência é particularmente crítico. Em grandes turmas que podem ter centenas de alunos e ter vários monitores e/ou professores é difícil unificar critérios de correção. Em trabalhos acadêmicos complexos envolvendo estudos empíricos, em níveis de graduação, mestrado e doutorados onde a adoção e avaliação dos critérios são cada vez mais complexos.

Foi constatado a necessidade de consolidar informações sobre que tipo de rubricas foram propostas por diversos pesquisadores

nos estudos empíricos desenvolvidos por eles nos artigos selecionados na RSL deste artigo.

Os resultados obtidos foram uma nova categorização dos tipos de rubricas que estão descritos na Tabela 4 e serão discutidos na seção 7.

Tabela 4: Categorização rubrica baseada nos resultados empíricos dos artigos selecionados na RSL

Tipo de rubrica categorizada empiricamente	Porcentagem deste resultado [artigo analisado]
Adaptativa	4,35 % [18]
Avaliativas	39,13% [5, 6, 8, 9, 14, 17, 22, 25, 27]
Formativas	9,52% [21, 23]
Múltiplas	9,52% [10, 24]
Pedagógicas	9,52% [7, 15]
Metodológicas (Métodos de Pesquisa)	4,35 % [16]
Padronizáveis para fornecer análise quantitativa	19,05% [13, 19, 20, 26]
Total	100%

Rubricas adaptativas são rubricas avaliativas customizadas pela definição e coleta de metadados que auxiliem a extrair evidências da aprendizagem do aluno. Os metadados referem-se a determinados itens da rubrica avaliativas cujo objetivo é adaptar informações relevantes sobre os limites de decisão usados na avaliação docente para classificar as atividades discentes [18].

Rubricas avaliativas são as mais comuns usadas por pesquisadores nos artigos selecionados na RSL deste artigo, boas rubricas avaliativas devem definir critérios de avaliação, de qualidade e uma estratégia de pontuação dos itens da rubrica [3].

Rubricas formativas se concentram em aspectos do processo formativo do estudante escolhido pelo pesquisador que vai além da aquisição de conhecimentos intrínsecos de uma disciplina, deve-se também prosseguir com o desenvolvimento do pensamento crítico discente, de forma a gerar novos aprendizados que os integrem com a ética e as responsabilidades do cotidiano [21].

Rubricas múltiplas aparecem em pesquisas que usam mais de um tipo de rubrica onde cada uma delas é usada para mensurar diferentes aspectos do ensino ou da aprendizagem no contexto de sala de aula, por exemplo, com foco na interação dos estudantes e na construção de conhecimento multicultural [24].

Rubricas metodológicas apesar de raras são uma tendência inovadoras e representam o uso de rubrica como um método de pesquisa científica propriamente dita ou como parte fundamental para o desenvolvimento da metodologia de pesquisa, por exemplo,

rubricas usadas na amostragem de informações para a inferência em *Deep Learning* com vista a produção de *feedbacks* [16].

Rubricas padronizáveis são rubricas avaliativas construídas com objetivo de apoiar o processo de análise quantitativa dos dados em populações cujas as amostras estão em distribuição amostral de uma estatística específica, por exemplo, o desempenho de estudantes em questionários pode ser pontuado objetivamente usando uma rubrica padronizada para este fim [19].

6 Limitações da RSL

A RSL desenvolvida na elaboração deste artigo tem como base uma ferramenta robusta para conduzir sólidas revisões sistemáticas com um protocolo robusto, seguido com rigor científico, entretanto o processo de pesquisa possui limitações. Nesta seção será discutido algumas delas, a saber:

- Exclusão de algum trabalho relevante que esteja em outro idioma que não o inglês que sejam entendíveis pelos autores deste artigo como o português e o espanhol.
- Exclusão de uma base relevante de busca no idioma inglês disponível da plataforma CAPES e não utilizados por nós devido às limitações de tempo e espaço.
- Exclusão de artigos relevantes que podem não ter sido incluídos nos critérios de inclusão definidos no protocolo (seção 3.2.2);
- Decisões subjetivas dos revisores, em razão da exclusão de documentos que não apresentaram uma descrição clara do uso da rubrica do ponto de vista dos revisores, autores deste artigo.

Apesar das limitações inerentes as escolhas de quaisquer RSL foram possíveis mostra resultados importantes que permite obter um panorama significativo sobre pesquisas contidas nas três bases pesquisadas envolvendo rubricas e aprendizagem de programação.

7 Discussão dos Resultados e Conclusões

A ferramenta Parsifal realmente auxilia a condução de sólidas RSLs, seu uso é fortemente recomendado ou ela ou outras ferramentas para RSL que estejam apoiadas em metodologias consistentes com é o caso do Parsifal.

A Tabela 1 consolida o resultado da busca, forma aceitos 10,72% dos artigos, 61,73% foram rejeitados e 27,55% foram duplicados. O que nos permite concluir que as pesquisas envolvendo rubricas e aprendizagem são bem maiores em outros contextos, aproximadamente 62% e somente ao redor de 11% são pesquisas envolvendo rubricas na aprendizagem de programação de acordo os resultados da RSL.

Nestes quase 62% de artigos rejeitados pelos nossos critérios de exclusão observar-se que existem estudos empíricos que incluem pesquisas em Informática e Educação, mas não foram avaliadas pois o foco do artigo é rubricas em aprendizagem de programação. Artigos usando algum tipo de linguagem de programação ou em disciplinas com ferramentas exclusivas, por exemplo, Sistemas Operacionais. É importante citar que existem trabalhos em áreas de pesquisa em Informática e Educação, que estão fora do escopo deste artigo, mais podem ser o campo de pesquisa de outros pesquisadores, a saber, artigos baseados em games, baseado em robótica, estudos educacionais que envolvam rubricas mais

aplicados a outras áreas como saúde e matemática, estudos envolvendo aprendizagem de programação em K-6 ou K-12.

O tema tem um grande potencial de crescimento em aprendizagem de programação, bem como no ensino, principalmente em cenário onde o ensino é à distância ou que tenham disciplinas à distância que tende a se tornar ainda mais relevantes em um futuro próximo. Não é à toa que AVAs importantes como *Moodle*, *Google Classroom*, *Blackboard* tem fornecidos suporte a rubricas nos últimos anos, embora conforme foi constatado, os professores das linguagens de programação não exploram o potencial da rubrica pré-definidas ou especializadas ou mesmo como um método de pesquisa pedagógica propriamente dito. À medida que pesquisas como a descrita neste *paper* forem mais conhecidas certamente haverá um incremento no tema.

A Figura 4 nos mostra que as pesquisas envolvendo rubricas e aprendizagem de programação são crescentes, os primeiros datam do ano de 2003, mas o interesse na área vem crescendo entre 2019 e 2020.

A padronização da extração usando o *Data Extraction Form* do Parsifal nos permitiu extrair informações fundamentais para RSL conforme foi descrito na Tabela 2.

A opção dada pela ferramenta Parsifal para a extração de dados gerados em uma planilha que pode ser aberta, por exemplo, no LibreOffice Calc ou no Microsoft Excel demonstrou-se bastante útil pois permitiu obter as estatísticas percentuais nas respostas a QP1 e QP2.

A rigor quaisquer categorizações descritas em respostas as perguntas de QP1 ou QP2 poderiam gerar uma tabela como foi feito na Tabela 3 com a categorização das linguagens de programação usadas pelos pesquisadores e com a Tabela 4 com a categorização rubrica baseada nos resultados empíricos. Foi usado as categorizações das linguagens de programação e do tipo de rubrica porque elas são apropriadas ao escopo deste artigo.

A rubrica foi usada em estudos empíricos em sala de aula em 85,72% das pesquisas e em 14,28% não foi usada. Analisando os números existe uma forte tendência para rubricas ser usada, em futuras pesquisas, também no contexto de sala de aula. O que é bastante promissor pois seu uso pode melhorar a padronização das avaliações e a produção de *feedbacks* mais específicos.

A Tabela 3 indica que 23,81 % das pesquisas são em disciplinas do tipo CS1, quase $\frac{1}{4}$ das pesquisas o que é significativo uma vez que nestas disciplinas as ferramentas do tipo *Online Judge* são hegemônicas, porém o uso de rubrica como ferramenta de avaliação está aumentando. Em CS2 19,05% dos pesquisadores usaram rubricas que demonstra a utilidade das rubricas para mapear conhecimentos típicos da Orientação a Objetos como polimorfismo e herança.

A Tabela 4 é uma contribuição relevante porque os tipos de rubricas podem variar entre autores e é difícil fazer uma categorização consistente em uma quantidade massiva de rubricas criadas apenas para atender uma determinada necessidade pontual. Este fenômeno ocorreu nos milhares tipos de rubricas cadastradas na ferramenta RubStar que, muitas vezes, replicam as rubricas porque os usuários da ferramenta só podem manipular as rubricas que tem permissão de escrita.

Um objetivo pedagógico deve ser sempre passível de avaliação, pois senão não a como avalia-lo concretamente. Acontece, entretanto, que a variedade e a complexidade das expectativas de aprendizagem demandam instrumentos de avaliação diferentes. Se este instrumento for uma rubrica a categorização dos tipos de

rubricas usadas em pesquisas pode indicar um caminho a novos pesquisadores.

Na Tabela 4 foi consolidado sete tipos de rubricas efetivamente usadas por pesquisadores: adaptativa (4,35%), avaliativas (39,13%), formativas (9,52%), múltiplas (9,52%), pedagógicas (9,52%), metodológicas que são usadas como métodos de pesquisa científica (4,35%) e padronizáveis para fornecer análise quantitativa (19,05%).

O tipo de rubrica mais popular entre os pesquisadores é a avaliativa dada a necessidade de se avaliar os estudantes. As padronizáveis servem para fornecer análise quantitativa, formativas e as múltiplas (mais de um modelo de rubrica usado no estudo) também podem servir para este fim, mas podem construir *feedback* mais ricos semanticamente.

As rubricas adaptativas e metodológicas bem como as usadas como método de pesquisas ainda são poucos exploradas.

As rubricas usadas como métodos de pesquisas são potencialmente mais promissoras pois podem suportar pesquisas bem mais complexas envolvendo diversas abordagens metodológicas, tais como, suporte a variados métodos de pesquisa, qualitativos e quantitativos.

As pesquisas referentes a rubricas descritas na seção 2 foram necessárias pois nos permitiu avaliar as pesquisas especificamente rubricas na aprendizagem de programação de forma livre, sem o formato da extração de dados da ferramenta Parsifal.

Analisando estas pesquisas e as respostas de QP1 e QP2 conclui-se que a rubrica pode ser usada como um método de pesquisa consistente para padronizar e mensurar a aprendizagem discentes com critérios explícitos permitindo o mapeamento de objetivos pedagógicos de uma determinado exercício, tarefa ou disciplina usando diversos tipos de tipo de rubrica categorizada empiricamente na Tabela 4.

A RSL permitiu observar que há preocupações sobre a falta de distinção dentro da estrutura do mapeamento do conteúdo e dos níveis de avaliação da rubrica, neste caso, os autores decidiram adaptar a rubrica de acordo com critérios teóricos e empíricos da pesquisa deles.

Os estudantes são mais propensos a ver as rubricas como ferramentas de instrução. Os professores são mais propensos a ver as rubricas como ferramentas de avaliação. Seja qual for o caso pode existir uma contribuição significativa em rubricas, principalmente nos modelos formais da Tabela 4.

Sendo assim, o uso de rubricas pode contribuir para o professor mapear eventuais falhas na aprendizagem não triviais e definir melhores estratégias ao ensino com base no desempenho discente.

AGRADECIMENTOS

Agradecemos a Universidade Federal de Goiás (UFG) e o apoio dado a pesquisa pelo Programa de Pós-Graduação em Ciência da Computação do Instituto de Informática (INF) da UFG.

REFERÊNCIAS

- [1] Dannelle D. Stevens, Antonia J. Levi, Barbara E. Walvoord, 2013. Introduction to Rubrics: An Assessment Tool to Save Grading Time, Convey Effective Feedback, and Promote Student Learning. Second Edition. Stylus Publishing, LLC. ISBN 978-1-57922-587-2.
- [2] Alistair Campbell, 2005. Application of ICT and Rubrics to the Assessment Process Where Professional Judgement Is Involved: The Features of an e-Marking Tool. *Assessment & Evaluation in Higher Education* 30 (5):529-537. DOI: 10.1080/02602930500187055.
- [3] Malini Y. Reddy and Heidi Andradeb, 2010. A Review of Rubric Use in Higher Education. *Assessment & Evaluation in Higher Education*, vol. 35, no. 4, pp. 435-448, 2010. DOI: 10.1080/02602930902862859.
- [4] RubStar - Create Rubrics for your Project-Based Learning Activities. Disponibilizada em <http://rubistar.4teachers.org/index.php>. Acessada em setembro de 2020.
- [5] Al-Khalifa K. Amal and Marie Devlin, 2020. Evaluating a Peer Assessment Approach in Introductory Programming Courses. In *United Kingdom & Ireland Computing Education Research conference*. (UKICER '20). ACM Press, New York, NY, USA, 51-58. DOI: <https://doi.org/10.1145/3416465.3416467>
- [6] Xihui Zhang, John D. Crabtree, Mark G. Terwilliger & Tyler T. Redman, 2020. Assessing Students' Object-Oriented Programming Skills with Java: The "Department Employee" Project, *Journal of Computer Information Systems*, 60:3, 274-286. DOI: 10.1080/08874417.2018.1467243
- [7] Dorodchi M., Dehbozorgi N., Frevert T.K., 2017. "I wish I could rank my exam's challenge level!": An Algorithm of Bloom's Taxonomy in teaching CS1. *Proceedings - Frontiers in Education Conference, FIE, 2017-October*, pp. 1-5. DOI: 10.1109/FIE.2017.8190523
- [8] Sebastian Garces; Guity Ravai; Camilo Vieira; Alejandra J. Magana, 2019. Effects of Self-explanations as Scaffolding Tool for Learning Computer Programming. *IEEE Frontiers in Education Conference (FIE), Covington, KY, USA, 2019*, pp. 1-6, doi: 10.1109/FIE43999.2019.9028561.
- [9] Tuukka Ahoniemi, Essi Lahtinen, and Tommi Reinikainen. 2008. Improving pedagogical feedback and objective grading. *SIGCSE Bull.* 40, 1 (March 2008), 72-76. DOI:<https://doi.org/10.1145/1352322.1352162>.
- [10] Nilima Salankar, 2019. Impact of Rubrics, Addie and Gagne model on the performance of students in programming subject. *International Journal of Engineering and Advanced Technology (IJEAT)*. ISSN: 2249 - 8958, Volume-9 Issue-1, October 2019.
- [11] Barbara Kitchenham and Stuart M. Charters, (2007). Guidelines for Performing Systematic Literature Reviews in Software Engineering (EBSE 2007-001). Keele University and Durham University Joint Report.
- [12] Torrey Trust and Emrah Pektas, 2018. Using the ADDIE Model and Universal Design for Learning Principles to Develop an Open Online Course for Teacher Professional Development. *Journal of Digital Learning in Teacher Education*, 34(4), 219-233.
- [13] Maknuna, Rafiq Durotul et al., 2017. "Effectiveness of Use of Technical Skill Assessment Instruments to Increase Web Programming Competency." (2017). DOI:10.2991/ICOVET-17.2017.9.
- [14] Osvaldo Clúa and Maria Feldgen, "A first course in Operating Systems with and without rubrics," 2011 *Frontiers in Education Conference (FIE)*, Rapid City, SD, 2011, pp. F3D-1-F3D-5, doi: 10.1109/FIE.2011.6142820.
- [15] Dirson S. Campos, Antônio J. Mendes, Maria J. Marcelino, Deller J. Ferreira and Lenice M. Alves, 2012. A Multinational Case Study on using Diverse Feedback Types Applied to Introductory Programming Learning. *42nd Annual Frontiers in Education Conference (FIE)*; Seattle, WA; United States; 3 October 2012.
- [16] Mike Wu, Milan Mosse, Noah Goodman, Chris Piech, 2019, Zero Shot Learning for Code Education: Rubric Sampling with Deep Learning Inference. *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, 782-790. ISSN 2374-3468 (Online), Volume 33 Number 1 (2019).
- [17] Katrin Becker, 2003. Grading programming assignments using rubrics. *ACM Sigcse Bulletin*. 35. 253. 10.1145/961290.961613.
- [18] Marco Carmosino and Mia Minnes. 2020. Adaptive Rubrics. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education (SIGCSE '20)*. Association for Computing Machinery, New York, NY, USA, 549-555. DOI:<https://doi.org/10.1145/3328778.3366946>.
- [19] Shanon M. Reckinger and Bryce E. Hughes. "Measuring Differences in Performance by Varying Formative Assessment Construction Guided by Learning Style Preferences." (2017).
- [20] Shanon M. Reckinger, Bryce E. Hughes. 2018. Partnering Strategies for Paired Formative Assessment in Programming. *Conferência American Society of Engineering Education Annual Conference & Exposition*.
- [21] Norka Bedregal-Alpaca, Jaén Doris S. Tupacyupanqui and Víctor Manuel C. Aparicio, 2019. The TPACK model as the basis of a didactic proposal for the teaching-learning of Linear Programming. *IEEE World Conference on Engineering Education (EDUNINE)*, Lima, Peru, 2019, pp. 1-6, doi: 10.1109/EDUNINE.2019.8875763.
- [22] Srimadhaven T, Chris Junni AV, Naga Harshith, Jessenth Ebenezer S, Shabari Girish S, Priyaadharshini M, 2019. Learning Analytics: Virtual Reality for Programming Course in Higher Education. *9th World Engineering Education Forum, WEEF 2019*. *Computer Science* 172 (2020) 433-437.
- [23] Dianne Raubenheimer, Jeff Joines and Ami Craig, 2009. Using Computational Tools To Enhance Problem Solving. *Annual Conference & Exposition, Austin, Texas*. DOI:10.18260/1-2-4610.

- [24] Gihan Osman, Susan C. Herring, 2007. Interaction, facilitation, and deep learning in cross-cultural chat: A case study. *The Internet and Higher Education* 10 (2007) 125–141. DOI: <https://doi.org/10.1016/j.iheduc.2007.03.004>
- [25] Kristina von Hausswolff and Anna Eckerdal, "Measuring Programming Knowledge in a Research Context," 2018 IEEE Frontiers in Education Conference (FIE), San Jose, CA, USA, 2018, pp. 1-9. DOI: 10.1109/FIE.2018.8658615.
- [26] Marcos Gestal, Carlos Fernandez-Lozano, Cristian R. Munteanu, Juan R. Rabuñal and Julian Dorado, 2018. Evaluation as a Continuous Improvement Process in the Learning of Programming Languages. In: Graña M. et al. (eds) *International Joint Conference SOCO'18-CISIS'18-ICEUTE'18*. SOCO'18-CISIS'18-ICEUTE'18 2018. *Advances in Intelligent Systems and Computing*, vol 771. Springer, Cham. https://doi.org/10.1007/978-3-319-94120-2_50
- [27] Sook-Young Choi, 2019. Development of an Instructional Model based on Constructivism for Fostering Computational Thinking. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* ISSN: 2278-3075, Volume-8 Issue-3C, January 2019.
- [28] Joseph, S. Rickett, C. Northcote M, Christian, B. J. (2020). "Who are you to judge my writing?": Student collaboration in the co-construction of assessment rubrics. *New Writing*, 17:1, 31-49, DOI: 10.1080/14790726.2019.1566368.
- [29] David Vaccari and Siva Thangam, 2010. A Proposed Doctoral Assessment Procedure And Rubric For Science And Engineering Paper presented at 2010 Annual Conference & Exposition, Louisville, Kentucky. DOI: 10.18260/1-2—16106.
- [30] Gerriet Janssen, Valerie Meier, Jonathan Trace, 2015. Building a Better Rubric: Mixed Methods Rubric Revision. *Assessing Writing*, Volume 26, 2015, Pages 51-66, ISSN 1075-2935, <https://doi.org/10.1016/j.asw.2015.07.002>.