

# Projeto e Desenvolvimento de um Software de Apoio à Revisão Aberta por Pares

Matheus A. Pereira, Nelson Tenório, Edson Oliveira Jr

<sup>1</sup>Departamento de Informática – Universidade Estadual de Maringá (UEM)  
Maringá – PR – Brasil

matheus.add12@gmail.com, nntjunio2@uem.br, edson@din.uem.br

**Resumo.** *A Ciência Aberta (CA) pode significar, desde práticas até iniciativas de pesquisa, com o objetivo de tornar o conhecimento acessível a toda a comunidade de interesse. Seus princípios incluem dados abertos para disseminação e transparência, acesso aberto para transferência mais rápida de conhecimento através de trabalho compartilhado, código aberto para o fácil compartilhamento de dados e ferramentas, métodos abertos para desenvolvimento de padrões para protocolos de coleta e metadados, recursos abertos para treinamento em métodos CA e, finalmente, revisão por pares aberta para alcançar rigor científico. Um dos pilares do AC é o processo Open Peer Review (OPR), um derivado do Peer Review (PR). Neste artigo apresentamos o PeerVise, protótipo de plataforma colaborativa que dá suporte ao OPR. O protótipo foi desenvolvido seguindo as boas práticas de Engenharia de Software e coletamos feedback de especialistas para validá-lo. Os resultados mostraram percepções positivas quanto à proposta do protótipo, facilidade de uso e sugestões de melhorias.*

**Abstract.** *Open Science (OS) meanings practices to research initiatives, with a common goal: to make knowledge accessible to the entire community of interest. The main principles of OS include open data for better dissemination and transparency, open access for faster knowledge transfer through shared work, open source for easier data and tool sharing, open methods for developing standards for collection protocols and metadata, open resources for better training in OS methods, and finally, open peer review to achieve greater scientific rigor. One of the pillars of OS is the Open Peer Review (OPR) process, a derivative of Peer Review (PR). So, here we present the PeerVise, a collaborative platform prototype that supports the OPR process. The prototype was developed following the best practices of Software Engineering. We collected feedback from experts to validate the prototype. The results showed positive perceptions regarding the prototype's utility, ease of use, and improvement suggestions.*

## 1. Introdução

A Ciência Aberta (CA) engloba significados desde práticas até iniciativas para pesquisas, com o objetivo de tornar o conhecimento acessível à toda a comunidade de interesse, com ampla visibilidade dada a recomendação da Organização das Nações Unidas para a Educação, Ciência e Cultura (UNESCO) em novembro de 2021. Um dos pilares da CA é o processo de *Open Peer Review* (OPR), derivado do *Peer Review* (PR) [Ross-Hellauer 2017]. Essa prática intensifica a disseminação de informação e deixa

aberto o processo de revisão de um artefato, que vai desde a identificação dos autores e revisores até o processo de aceitação ou rejeição da submissão.

O processo de OPR permite que todo o processo de revisão por pares seja feito de forma transparente e colaborativa em contraponto ao modo privado e sigiloso. Isso permite que os revisores compartilhem seus comentários a respeito de um artefato abertamente. Alguns dos periódicos que utilizam o processo de revisão aberta aos pares, para aumentar a confiança na revisão, são: *PeerJ2*, *F1000Research3* e *Open Library of Humanities*. Os objetivos deste artigo foram especificar e implementar, de modo preliminar, um protótipo de software para apoio a OPR nomeado inicialmente de *PeerVise*, buscando unir as palavras *peer* e *advise*, sugerindo a ideia de uma plataforma colaborativa. O principal objetivo é deixar todo o processo de uma submissão disponível de forma pública para qualquer interessado no assunto, desde a identificação do autor e dos revisores, até a publicação dos relatórios com os resultados para a respectiva submissão.

A fim de alcançar os objetivos propostos efetuou-se duas etapas, a primeira delas destinada à coleta de informações relacionadas ao OPR e trabalhos relacionados. Na mesma etapa, foram definidos os requisitos e os casos de uso, a modelagem dos diagramas de classes referentes e a modelagem dos seus diagramas de pacotes. A segunda etapa foi destinada à implementação de um protótipo funcional de acordo com as melhores práticas da Engenharia de Software.

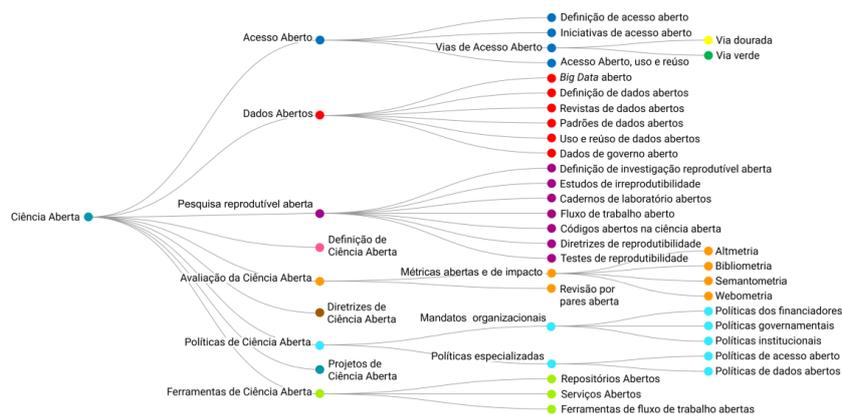
Por fim, este trabalho se estrutura com a seção 2 que apresenta uma revisão literária com os principais conceitos a respeito do objeto de estudo; a seção 3 que apresenta os requisitos e a modelagem do banco de dados seguida dos casos de uso, do protótipo e as suas etapas de desenvolvimento, o planejamento do questionário; a seção 4 apresenta a análise e discussão dos resultados; e a seção 5 apresenta as considerações finais seguida das referências bibliográficas.

## **2. Ciência Aberta**

A Ciência Aberta (CA) é definida por [Vicente-Saez and Martinez-Fuentes 2018] como um conhecimento transparente e acessível, partilhado e desenvolvido por meio de redes de colaboração que se utilizam de diferentes práticas, buscando seguir o princípio de que o processo científico deve ser compartilhado, sendo transparente e disponível ao público, como definido por *OpenScienceMooc* [MOOC 2019], Figura 1. A figura apresenta ramificações da ciência aberta, como o acesso aberto, que visa à disponibilização de informações científicas, publicações acadêmicas, livros, dados, códigos, metadados e relatórios de pesquisa de forma gratuita e online.

### **2.1. Revisão por Pares**

A Revisão por Pares (*Peer Review* - PR), [Alberts et al. 2008], é o processo no qual especialistas em determinada área examinam e criticam resultados científicos apresentados por um determinado autor, antes da publicação deles em algum meio de comunicação. O objetivo é garantir que os resultados das pesquisas sejam confiáveis e precisos, assegurando a qualidade e a relevância do trabalho científico. Além disso, a revisão visa garantir a originalidade e a rigorosidade da pesquisa, bem como a conformidade do estudo com as normas e padrões estabelecidos pela comunidade científica.



**Figura 1. Taxonomia aberta. Fonte: [Vicente-Saez and Martinez-Fuentes 2018]**

Alguns tipos de PR mais utilizados são: *single blind review*: o autor não tem conhecimento algum da identidade do revisor e os revisores possuem conhecimento da identidade dos autores; *double blind review*: nem o revisor nem o autor tem conhecimento sobre a identidade um do outro; *triple blind review*: adiciona uma nova camada de anonimato, ocultando, além da identidade do autor e do revisor, a identidade do editor responsável pela submissão; e o *open peer review*: a identidade do autor, revisor e também a do editor é reconhecida por todos os envolvidos durante todo o processo.

## 2.2. Revisão Aberta por Pares

O processo de OPR, apesar de ter seu uso aumentado de forma considerável desde a virada do século, representa um dos últimos aspectos do movimento da CA a ser amplamente adotado [Wolfram et al. 2020].

OPR é um processo que depende e incentiva a confiança mútua, o respeito e também uma abertura à crítica [Schmidt et al. 2018]. Em um processo de revisão, autores e revisores podem concordar ou discordar em alguns aspectos da submissão e expor isso de forma pública é essencial, ainda de acordo com os autores, para que os revisores tenham ciência de que comunicaram seus pontos de uma forma clara e civilizada para o autor, de forma que tais pontos, sirvam realmente como um *feedback* valioso.

Um estudo realizado sobre os efeitos do OPR na qualidade da revisão [Van Rooyen et al. 1999], mostrou que revisões feitas com o processo, não tiveram nenhum efeito negativo na qualidade, entretanto, a probabilidade de revisores se recusarem a revisar uma determinada submissão, aumentou de forma significativa. [Nascimento and Albagli 2019], mostrou a aceitação positiva dos revisores em terem suas identidades reveladas, e isso viabilizou a implantação do processo de OPR. Também foi concluído que revisões realizadas seguindo esse processo são pelo menos tão boas quanto revisões que utilizaram outros processos de revisão disponíveis, apesar de poderem levar mais tempo para serem concluídas, o que segundo [Nascimento and Albagli 2019] é um fator positivo, pois, os revisores têm uma carga maior de responsabilidade ao realizar apontamentos em uma submissão quando seus nomes são publicados abertamente, o que favorece a qualidade do processo editorial.

Um sistema pensado e desenvolvido por [Choi and Seo 2021], mostra o processo da implementação do OPR no sistema da JISTAP (*Journal of Information Science The-*

*ory and Practice* que foi realizada utilizando uma rede privada de blockchain baseada em *smart contracts*. A implementação conta com um modelo de aceitação gradual do processo de OPR, permitindo que os autores selecionem apenas os aspectos que se sentem confortáveis nessa forma de revisão. Uma iniciativa encontrada na literatura sobre o processo de OPR é referente a *F1000 Research*, que segundo [Kirkham and Moher 2018], é uma plataforma alternativa para publicação de artigos, oferecendo tipos de PR e entre eles o OPR. Os autores concluíram que a maioria das pessoas que publicaram na ferramenta tiveram uma experiência positiva e publicariam novamente na plataforma, entretanto, foram apresentadas algumas preocupações sobre a qualidade dos arquivos publicados.

### 3. Materiais e Métodos

Com a finalidade de executar a pesquisa, seguiu-se o protocolo apresentado na Figura. Tal figura apresenta o fluxo das etapas de efetuadas nesta pesquisa. As etapas seguiram as boas práticas da Engenharia de Software. Como os requisitos eram bem definidos e conhecidos e, por isso, foi utilizado um processo de software clássico.

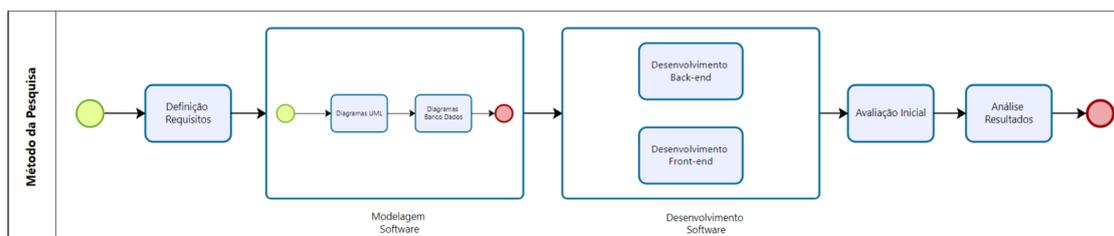


Figura 2. Diagrama de casos de uso.

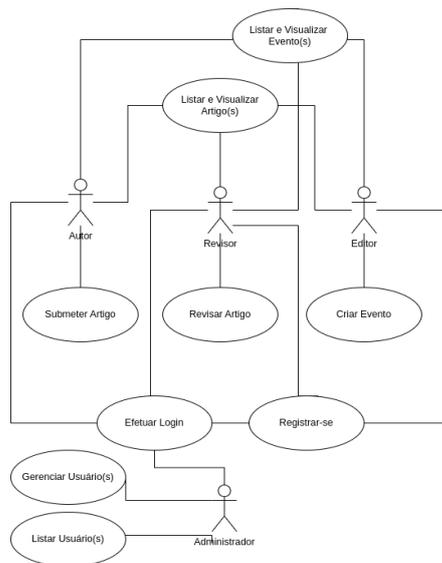
#### 3.1. Definição dos Requisitos

Como principais requisitos a serem desenvolvidos, têm-se divididos em Requisitos Funcionais (RF) que definem o que o software desenvolvido deve fazer e Requisitos não Funcionais (RNF) definem as características do sistema, como desempenho e usabilidade. Um resumo dos requisitos funcionais do protótipo são apresentados a seguir: Os requisitos são: '**RF.1** (essencial)': Login no sistema. '**RF.2** (essencial)': Cadastrar usuário no sistema, o cadastro pode ser realizado por qualquer pessoa e por padrão a permissão selecionada é de autor, podendo consultar eventos e artigos, além de poder realizar a submissão de um novo artigo. Entretanto, pode ser realizado a alteração da permissão durante o cadastro e não poderá ser alterado após o cadastro. '**RF.3** (importante)': Editar dados de um usuário previamente cadastrado. '**RF.4** (desejável)': Permitir ao administrador ativar ou desativar um usuário. '**RF.5** (desejável)': Permitir a listagem de todos os usuários cadastrados ou de um usuário específico. '**RF.6** (essencial)': Cadastrar um novo evento. '**RF.7** (essencial)': Listar todos os eventos cadastrados ou um evento específico. '**RF.8** (essencial)': Listar todos os artigos cadastrados ou um artigo específico. '**RF.9** (essencial)': Permitir a submissão de um artigo para um respectivo evento preenchendo um breve formulário. O usuário pode efetuar novas submissões do artigo. '**RF.10** (essencial)': Permitir a revisão de um artigo submetido para um determinado evento. '**RF.11** (importante)': Permitir o *logout* do sistema. Os requisitos não-funcionais foram o sistema: '**RNF.1** (essencial)': ser seguro, protegendo os dados informados; '**RNF.2** (essencial)': ser capaz de suportar usuários simultâneos sem atrasos ou travamentos; '**RNF.3**

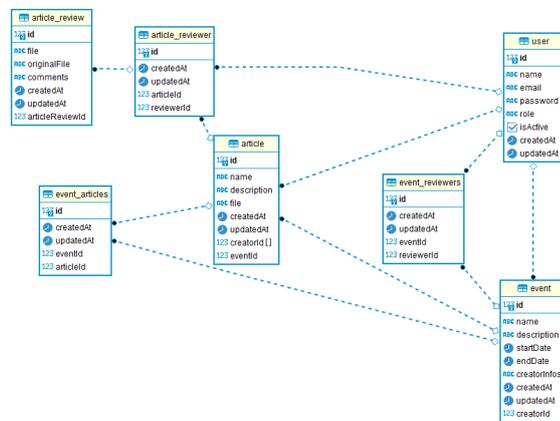
(essencial)’: ser fácil de utilizar e acessível aos seus usuários; **RNF.4** (importante)’: ser compatível com diferentes navegadores; **RNF.5** (desejável)’: ter um menu principal.

### 3.2. Modelagem do Software

A Figura 3.a apresenta o diagrama de caso de uso modelado com base nos requisitos levantados. Diagrama de caso de uso que apresenta as principais funcionalidades e atribuições presentes no sistema, sendo: "cadastro de usuário"; "login"; "visualizar artigos"; "visualizar eventos"; "submeter artigo"; "revisar artigo"; "gerenciar eventos" e ; "gerenciar usuários".



a. Caso de Uso



b. Modelo do Banco de Dados

Figura 3. Modelagens de Caso de Uso e Banco de Dados.

### 3.3. Modelagem do Banco de Dados

No desenvolvimento do sistema efetuou-se a criação de um banco de dados relacional no PostgreSQL na versão 15.1 e a ferramenta DBEAVER para visualização do *schema*, entidades e dados. O diagrama entidade-relacionamento, Figura 3.b, inclui sete entidades, sendo: *article-review*, contém os dados de uma revisão submetida por um revisor; *article-reviewer*, possui duas chaves estrangeiras, sendo a primeira para identificar o artigo revisado e a segunda para identificar o revisor; *article*, tendo os dados de um artigo submetido; *event-articles*, possui duas chaves estrangeiras, sendo a primeira para identificar o evento ao qual o artigo foi submetido e a segunda para identificar o artigo submetido; *event-reviewers*, duas chaves estrangeiras, sendo a primeira que identifica o evento ao qual o revisor está inserido e a segunda que identifica o revisor; *event*, contendo os dados de um evento criado; *user*, contendo os dados de um usuário cadastrado.

### 3.4. Backend e Frontend

Para o desenvolvimento do *backend* foi escolhido o framework NestJS que fornece a estrutura de projeto baseada em arquitetura de microsserviços, permitindo dividir a aplicação em módulos, sendo cada um deles independentes e escaláveis. A Figura 4 apresenta uma das interfaces do sistema.

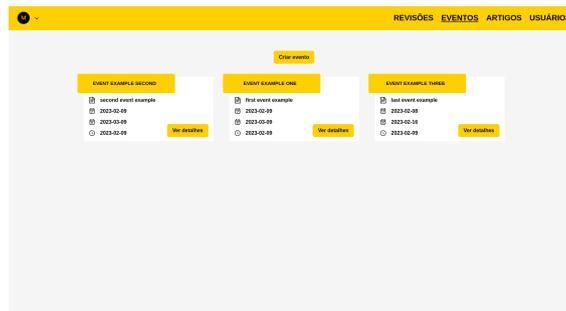


Figura 4. Interface principal do software.

### 3.5. Implantação

Para liberação e testes do sistema, foi realizada a implantação da aplicação. Para a implantação do banco de dados foi utilizada a ferramenta *Neon*, a ferramenta disponibiliza o banco de dados PostgreSQL. Para a implantação da aplicação, foi utilizada a ferramenta *Vercel* que efetua integrações com o Next.js para o desenvolvimento do sistema e hospedagem gratuita. Para a implantação do *backend* foi utilizada a ferramenta *Railway*, que disponibiliza de forma gratuita opções para hospedagem do *Nest.js*, utilizado na aplicação para o desenvolvimento *backend*.

### 3.6. Avaliação Inicial

Para a avaliação inicial do *PeerVise* desenvolveu-se uma pesquisa de opinião com base no *Technology Acceptance Model* (TAM), [Marangunic and Granic 2015]. A avaliação teve como objetivo obter informações dos participantes acerca da plataforma desenvolvida, analisando o nível de aceitação do sistema. As questões desenvolvidas, ora em forma de asserção ora em forma de pergunta, foram divididas em quatro aspectos. O primeiro foi Identificação do Participante (IP), o segundo a Utilidade Percebida (UP), o terceiro a Facilidade de Uso Percebida (FUP) e, por fim, um campo aberto para o participante deixar as suas impressões. A UP apresenta quanto um indivíduo acredita que o uso de um sistema pode melhorar seu trabalho e, quanto mais alta essa medida, mais os usuários acredita-se em uma relação positiva entre uso e desempenho. A FUP se refere a quanto a pessoa acredita que o uso de um sistema reduz esforços físicos ou mentais, ambas seções do questionário foram baseadas na sugestão de [Davis 1989].

#### 3.6.1. Planejamento

O questionário para avaliação do software é composto de asserções a serem respondidas com a escala Likert. As questões foram organizadas em quatro seções: IP, UP, FUP e questão aberta. Na primeira seção eram as questões de identificação do participante, *i.e.*, IP.1 Nome Completo; IP.2 E-mail; IP.3 Você já utilizou algum sistema de revisão por pares?; e IP.4 Se você marcou alguma opção "Sim" na questão anterior, informe o nome ou a URL do sistema utilizado; senão, digite "Não". A segunda seção (UP), investigou como o participante acreditava que o uso do software melhoraria o desempenho de suas tarefas. A seção foi composta das seguintes questões: **UP.1** O software desenvolvido me permite submeter artigos, criar e visualizar eventos, artigos e/ou revisões de outros usuários do

sistema de forma fácil e transparente. **UP.2** Visando o conceito de OPR, o software desenvolvido me permite ter um controle absoluto na escolha de revisores ao criar um evento, permitindo selecionar especialistas na área a qual o artigo se insere com base em sua experiência e conhecimento. **UP.3** Visando o conceito de OPR, o software desenvolvido me permite previamente de forma facilitada identificar os revisores do evento ao qual irei submeter um artigo. **UP.4** Visando o conceito de OPR, é possível ver com transparência as revisões enviadas para artigos de outros usuários dentro de um determinado evento. **UP.5** O software desenvolvido me permite de forma simples inserir um comentário de revisão para um artigo. **UP.6** O software desenvolvido me permite de forma simples identificar os comentários de revisão de um artigo submetido. **UP.7** No geral, considero o software desenvolvido útil no apoio à *open peer review*, fornecendo ferramentas úteis como submissão de artigos, criação de eventos e acompanhamento de outros artigos, eventos e/ou revisões dentro da aplicação. **UP.8** Visando melhorar a UP do software desenvolvido para apoio à OPR, sintá-se livre para escrever quaisquer sugestões para melhorias no software. A terceira seção referiu-se à FUP para entender o grau com que o participante acreditava que o uso do software envolvia baixo nível de esforço. A seção foi composta pelas questões: **FUP.1** O software desenvolvido é objetivo em suas tarefas. Normalmente, não fico confuso ao usá-lo. **FUP.2** O software desenvolvido me permite de forma simples visualizar um artigo, mensagens dos revisores e seu '.PDF' com as correções, correções do autor do artigo e toda interação com o editor. **FUP.3** As principais funcionalidades do software desenvolvido são de fácil acesso e não exigem conhecimento prévio para utilização. **FUP.4** A utilização do software desenvolvido exige muito do meu esforço. **FUP.5** A interação com o software desenvolvido é de fácil compreensão. **FUP.6** É fácil lembrar como realizar determinadas tarefas utilizando o software desenvolvido. **FUP.7** No geral, considero o software desenvolvido de fácil utilização. **FUP.8** Visando melhorar a FUP do software desenvolvido para apoio à OPR, sintá-se livre para escrever quaisquer sugestões para melhorias no software. A quarta seção possuía uma única asserção a respeito de sugestões e melhorias para o software desenvolvido, sendo: 'Com o objetivo de melhoria geral do software desenvolvido para apoiar a OPR, sintá-se à vontade para sugerir melhorias gerais que possam ser feitas no programa'.

### 3.6.2. Execução

Para a execução do estudo foi criado um formulário no Google Forms, (disponível em: <https://docs.google.com/forms>), que apresentava o Termo de Consentimento Livre e Esclarecido, uma descrição das principais funcionalidades da aplicação e as questões desenvolvidas para avaliação. Os participantes foram instruídos a ler os documentos fornecidos, acessar e utilizar o sistema. Somente ao final dessas ações os participantes responderam ao questionário.

## 4. Análise de Dados e Discussão

O formulário foi divulgado em um grupo de pessoas que tiveram contato com submissão de artigos para iniciação científica, mestrado ou doutorado em que sete aceitaram participar; seus perfis eram: quatro pessoas do gênero masculino e três feminino, todos experientes. Um desenvolvedor de software e o restante, professores. Somente um com graduação, o restante todos pós-graduados.

Para validar um resultado como positivo, considerou-se, ao menos, que metade dos participantes tenham respondido de forma positiva à afirmação.

A UP.1 questionou se o "software desenvolvido permite realizar as principais funcionalidades do sistema de forma fácil e transparente". Como resultado, 5 concordaram totalmente (71,4%) e 2 concordaram (28,6%). O resultado é positivo para as principais funcionalidades do software. A UP.2 questionou se o "software desenvolvido permite um controle absoluto na seleção dos revisores na criação de um evento por parte de um editor", 3 participantes concordaram totalmente (42,9%) e 4 concordaram (57,1%). O resultado é positivo em relação ao controle do editor sobre os revisores do evento. A UP.3 questionou se o "software desenvolvido permite a identificação prévia dos revisores de um determinado evento por parte dos autores" e 3 participantes concordaram totalmente (42,9%), 3 concordaram (57,1%) e 1 discordou (14,3%). O resultado foi positivo para a identificação dos revisores de um evento, mesmo com uma resposta negativa. A UP.4 questionou se o "software desenvolvido permite visualizar com transparência as revisões enviadas para artigos de outros usuários". Para essa pergunta, 3 concordaram totalmente (42,9%), 3 concordaram (57,1%) e 1 foi neutro (14,3%). A resposta indicou um bom resultado em relação a transparência das revisões enviadas para um artigo. A UP.5 questionou se o "software desenvolvido permite a inserção de comentários de revisão de forma simples". Somente 1 concordou totalmente (14,3%) e 4 concordaram (57,1%). Entretanto, obteve-se 1 resposta neutra (14,3%) e uma que discordava (14,3%). A UP.6 questionou se o "software desenvolvido permite identificar os comentários de revisores para um determinado artigo de forma simples". O resultado teve respostas 3 que concordaram totalmente (42,9%), 3 concordaram (57,1%) e 1 que discordara (14,3%). O resultado foi positivo em relação a identificação dos comentários dos revisores para um determinado artigo. E por fim, a UP.7 questionou se os "participantes consideravam o software desenvolvido útil para atividades de OPR". Obteve-se 4 que concordaram totalmente (57,1%) e 3 concordaram (42,9%). Constatou-se um resultado positivo em relação a utilidade do software para atividades ligadas a OPR. Mesmo com respostas positivas, as respostas neutras e negativas foram cuidadosamente estudadas nas questões abertas, assumiu-se como resultado positivo as respostas com ao menos metade dos participantes que responderam de forma positiva à afirmação, seja concordando totalmente ou concordando com a afirmação ou, especificamente na questão FUP.4, as respostas com discordância ou discordância total, dada a asserção negativa.

A FUP.1 questionou se o "software desenvolvido é objetivo em suas tarefas e permite o uso sem causar confusão" e obteve-se 3 respostas que concordaram totalmente (42,9%), 3 concordaram (42,9%) e 1 neutro (14,3%). O resultado indicou ser positivo em relação a objetividade das tarefas do sistema. A FUP.2 questionou se o "software desenvolvido permite de forma simples a interação com artigos, revisões e correções enviadas" e obteve-se 3 que concordaram totalmente (42,9%), 3 que concordaram (42,9%) e 1 que discordara (14,3%). O resultado foi considerado bom em relação a simplicidade ao executar as principais funcionalidades. A FUP.3 questionou se o "software desenvolvido permite sem um conhecimento prévio o acesso as funcionalidades desenvolvidas" e obteve-se 4 respostas que concordaram totalmente (57,1%), 2 que concordaram (28,6%) e 1 que discordara (14,3%). O resultado positivo se deu por boa parte dos usuários. A FUP.4 questionou se o "software desenvolvido exige muito esforço para utilização" e obteve-se 3 respostas que discordaram totalmente (42,9%), 3 que discordaram (42,9%) e 1 neutro

(14,3%). O resultado é considerado muito positivo em relação a facilidade para utilização do sistema por parte dos usuários. A FUP.5 questionou se a "interação com o software desenvolvido é de fácil compreensão" e obteve-se as seguintes 4 respostas que concordaram totalmente (57,1%), 2 que concordaram (28,6%) e 1 neutra (14,3%). O indicativo foi de bom resultado em relação a compreensão e interação com o usuário. A FUP.6 questionou se foi "fácil lembrar como realizar determinadas tarefas no software desenvolvido" e o resultado foi de 3 respostas que concordaram totalmente (42,9%), 3 que concordaram (42,9%) e 1 neutra (14,3%). Assim, o resultado foi considerado bom em relação à facilidade de lembrança para tarefas do sistema, apesar de uma resposta neutra que motivou uma busca nas questões abertas para melhor entendimento. A FUP.7 questionou se no "geral o software é de fácil utilização" e obteve-se as seguintes respostas: 4 concordam totalmente (57,1%), 2 concordam (28,6%) e 1 neutro (14,3%). O resultado indicou ser muito positivo em relação a facilidade de utilização do sistema. Mesmo com diversas respostas positivas, como ocorreu na UP, decidiu-se observar com detalhes as questões abertas. Nas questões abertas os participantes puderam responder livremente fazendo sugestões e apontar com detalhes as respostas negativas. A análise dessas questões foi efetuada por meio do agrupamento das sugestões semelhantes, sendo: i) melhoria na interface para facilitar a interação da pessoa com as funcionalidades do sistema, tornando as informações mais claras; ii) complemento de informações no cadastro do usuário, permitindo que cada um informe suas áreas de especialização e interesse, facilitando aos editores a seleção de revisores e identificando com clareza os especialistas na área; e iii) melhorias no processo de gerenciamento de artigos e no processo de revisão, facilitando a remoção ou negação de artigos submetidos que não eram do escopo ou da área do evento para o qual foram submetidos. Alguns pontos no processo de revisão como a identificação clara de comentários e opções para respostas individuais entre autores e revisores, também foram levantados pelos participantes da pesquisa.

#### **4.1. Ameaças à Validade**

As ameaças à validade deste estudo foram: i) à validade interna, onde a heterogeneidade dos participantes afeta os resultados já que eles podem ter diferentes níveis de conhecimento sobre o tema. Para reduzir isso, o questionário foi divulgado por uma semana no grupo de pesquisadores familiarizados com o assunto; ii) à validade externa, dificuldade de encontrar participantes adequados para o estudo limita o escopo da pesquisa. Para avaliar as funcionalidades os participantes foram selecionados com base nos seus conhecimentos, o que restringiu a diversidade das suas experiências, na maioria 'autores'. A terceira ameaça é quanto à validade de constructo, em que o questionário foi elaborado com questões que verificaram as principais funcionalidades do software. Por fim, apresenta-se as ameaças à validade da conclusão, em que a principal é o tamanho da amostra obtida, *i.e.*, sete participantes. Isso se demonstrou insuficiente para expressar a qualidade e eficiência do software de forma conclusiva. Assim, o software necessita ser avaliado por mais pessoas.

#### **4.2. Compartilhamento dos Dados do Estudo**

Os dados deste estudo sobre o projeto e implementação de software para apoio à *open peer review* em eventos científicos são compartilhados com base nos princípios da Ciência Aberta, por meio do website Zenodo, no seguinte endereço eletrônico: <https://zenodo.org/record/7765040#.ZBzLRtJKirw>.

## 5. Considerações Finais

Os principais objetivos desta pesquisa foi desenvolver o protótipo de um sistema para OPR denominado *PeerVise*. Seus requisitos foram validados por meio de um formulário eletrônico que considerou a UP, para identificar o grau em que o usuário enxerga como o sistema aumenta seu desempenho de trabalho, e à FUP, que identifica o grau em que cada usuário vê como o uso do sistema é livre de esforço. Com base em um formulário eletrônico contendo questões abertas e fechadas, identificou-se pontos de melhoria no sistema com uma 'boa' taxa de aceitação. Após a análise dos resultados, foi possível identificar também pontos de melhoria como a interface e alguns processos de gerenciamento. Embora um estudo incipiente, o sistema mostrou-se adequado para as funcionalidades a que se propõe, abrindo assim oportunidades para trabalhos futuros e novas funcionalidades como: envio de e-mails; incremento de dados cadastrais; alteração de permissões; criação de eventos; revisão de artigos sem a necessidade da criação de novas contas; melhorias e adições de requisitos funcionais e não funcionais. Portanto, o sistema desenvolvido é visto como um piloto de bem sucedido para o apoio à revisão aberta por pares.

## Referências

- Alberts, B., Hanson, B., and Kelner, K. L. (2008). Reviewing peer review. *Science*, 321(5885):15–15.
- Choi, D.-H. and Seo, T.-S. (2021). Development of an open peer review system using blockchain and reviewer recommendation technologies. *Science Editing*, 8(1):104–111.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, pages 319–340.
- Kirkham, J. and Moher, D. (2018). Who and why do researchers opt to publish in post-publication peer review platforms?-findings from a review and survey of f1000 research. *F1000Research*, 7.
- Marangunić, N. and Granić, A. (2015). Technology acceptance model: a literature review from 1986 to 2013. *Universal access in the information society*, 14:81–95.
- MOOC, O. S. (2019). Open science benefits. Accessed: 24/10/2023.
- Nascimento, A. and Albagli, S. (2019). Conceitos de ciência aberta no Brasil: uma revisão sistemática de literatura. *Enc. Nac. Pesq. em Ciência da Informação*, 20.
- Ross-Hellauer, T. (2017). What is open peer review? a systematic review. *F1000Research*, 6.
- Schmidt, B., Ross-Hellauer, T., van Edig, X., and Moylan, E. C. (2018). Ten considerations for open peer review. *F1000Research*, 7.
- Van Rooyen, S., Godlee, F., Evans, S., Black, N., and Smith, R. (1999). Effect of open peer review on quality of reviews and on reviewers' recommendations: a randomised trial. *BMJ*, 318(7175):23–27.
- Vicente-Saez, R. and Martinez-Fuentes, C. (2018). Open science now: A systematic literature review for an integrated definition. *J. Business Research*, 88:428–436.
- Wolfram, D., Wang, P., Hembree, A., and Park, H. (2020). Open peer review: promoting transparency in open science. *Journal of the Medical Library Association*, 108:19–19.