

JEMS

O SISTEMA DE GESTÃO DE SUBMISSÕES DA SBC



Marcelo Antonio Marotta
Carlos Raniery Paula dos Santos
Arthur Selle Jacobs
Marcus Vinícius Brito da Silva
Lisandro Zambenedetti Granville

JEMS

O Sistema de Gestão de Submissões da SBC

Marcelo Antonio Marotta

Carlos Raniery Paula dos Santos

Arthur Selle Jacobs

Marcus Vinícius Brito da Silva

Lisandro Zambenedetti Granville

Porto Alegre

Sociedade Brasileira de Computação

2026



Esta obra está sob a licença Creative Commons Atribuição 4.0 (CC-BY). Você pode redistribuir este livro em qualquer suporte ou formato e copiar, remixar, transformar e criar a partir do conteúdo deste livro para qualquer fim, desde que cite a fonte.

Dados Internacionais de Catalogação na Publicação (CIP)

J54 JEMS: o sistema de gestão de submissões da SBC
Marcelo Antonio Marotta, Carlos Raniery Paula dos Santos, Arthur Selle Jacobs, Marcus Vinícius Brito da Silva, Lisandro Zambenedetti Granville. – Porto Alegre : Sociedade Brasileira de Computação, 2026.
43 p. : il. : PDF ; 1 MB

Modo de acesso: World Wide Web.
ISBN: 978-85-7669-672-8 (e-book)

1. Sistemas de submissão. 2. Eventos científicos. 3. Comunicação científica. 4. Sistemas de informação. I. Marotta, Marcelo Antonio. II. Santos, Carlos Raniery Paula dos. III. Jacobs, Arthur Selle. IV. Silva, Marcus Vinícius Brito da. V. Granville, Lisandro Zambenedetti. VI. Sociedade Brasileira de Computação. VII. Título.

CDU 06.055.6

Ficha catalográfica elaborada por Annie Casali – CRB-10/2339
Biblioteca Digital da SBC – SBC OpenLib

1 Introdução

O *Journal and Events Management System* (JEMS) é o sistema Web da Sociedade Brasileira de Computação (SBC) que oferece suporte à seleção de artigos científicos para eventos e periódicos. Inicialmente voltado para as conferências e revistas da própria SBC, o JEMS passou, ao longo dos anos, a ser empregado por outras sociedades científicas e entidades (*e.g.*, IEEE, ACM, CAPES, Rede Nacional de Ensino e Pesquisa - RNP e Ministério da Ciência, Tecnologia, Informações e Comunicações - MCTI) em avaliações de outras naturezas (*e.g.*, seleção de projetos de pesquisa e avaliação de relatórios de P&D).

O JEMS foi implantado em 2002 e utilizado pela primeira vez no 21° Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC 2003), organizado em Natal, RN, sob a coordenação geral da Profa. Thais Batista e a coordenação do comitê de programa pelo Prof. Luiz Fernando Gomes Soares. Em 2004, o JEMS se tornou um sistema oficial da SBC, sendo adotado pela maioria dos eventos científicos nacionais em computação. Hoje, em 2026, 23 anos após sua implantação, o JEMS acumula uma base de mais de 5 mil eventos, mais de 650 mil revisões e mais de 300 mil autores cadastrados.

O JEMS evoluiu ao longo dos anos, chegando hoje à sua terceira versão, o **JEMS3**. Mais do que uma ferramenta administrativa, o JEMS tornou-se parte integrante da cultura de publicação científica em Computação no Brasil: coordenadores de eventos, revisores, autores e editores de periódicos organizam seus fluxos de trabalho em torno do sistema, que se consolidou

1 Introdução

como referência no ecossistema de ciência aberta da SBC. Essa centralidade confere ao JEMS uma responsabilidade particular: ser confiável, eficiente e capaz de evoluir sem interromper os processos dos quais depende.

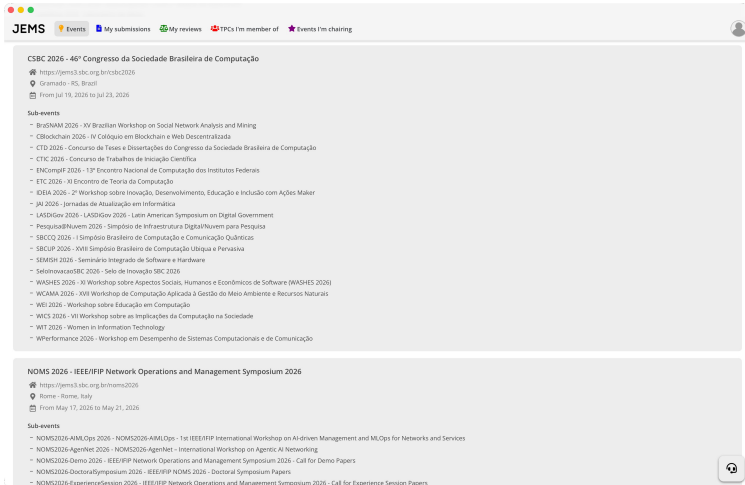


Figura 1.1: Tela inicial do JEMS3, exibindo a lista de eventos viáveis e seus respectivos sub-eventos para o usuário autenticado.

O JEMS3 não opera de forma isolada. Ele integra-se à SBC OpenLib (SOL), a biblioteca digital aberta da SBC, formando um *continuum* tecnológico que acompanha o artigo científico desde sua submissão até sua publicação permanente e indexação internacional. Essa integração, detalhada no Capítulo 6, representa um dos avanços mais significativos da versão atual.

Neste livreto, apresenta-se o histórico do JEMS e, em especial, a evolução tecnológica e as principais funcionalidades do

1 Introdução

JEMS3. O texto está organizado da seguinte forma: o Capítulo 2 narra a origem do sistema e sua evolução até o JEMS3; o Capítulo 3 descreve a arquitetura e a pilha tecnológica, incluindo uma linha do tempo detalhada do desenvolvimento; o Capítulo 4 apresenta os perfis de usuários e os fluxos de trabalho suportados; o Capítulo 5 descreve as principais funcionalidades por grupos temáticos; o Capítulo 6 detalha a integração com a SOL; e o Capítulo 7 traz as considerações finais sobre o impacto e o futuro do sistema.

Este livreto é, antes de tudo, um documento de memória institucional. O JEMS3 é um projeto construído por pessoas — pesquisadores que identificaram problemas, desenvolvedores que os resolveram, coordenadores de eventos que testaram e reportaram falhas, e membros da SBC que mantiveram vivo o compromisso com uma infraestrutura de qualidade para a ciência nacional. À medida que a comunidade cresceu — em número de eventos, pesquisadores e publicações —, o JEMS cresceu junto, adaptando-se a demandas que, em 2002, sequer eram imagináveis. Vale sublinhar que este não é um relato neutro: os autores são também protagonistas da história que nararam e escrevem com o olhar de quem conhece o sistema por dentro — seus pontos fortes, suas limitações e os dilemas que marcaram cada decisão de projeto. É com esse espírito que as páginas a seguir foram escritas.

2 Histórico e Evolução do JEMS

A trajetória do JEMS ao longo de mais de duas décadas é marcada por uma sequência de decisões técnicas e institucionais que moldaram o sistema tal como ele é hoje. Esta seção percorre esse histórico, desde a origem do sistema como solução para um problema pontual até sua consolidação como a principal plataforma de gestão de submissões da SBC.

2.1 O Problema Original e a Gênese do Sistema

O JEMS é resultado de uma ação para resolver um problema pontual no contexto da SBC: a necessidade de um mecanismo eficiente para a seleção de artigos para o SBRC 2003. O que surgiu como solução para um desafio específico evoluiu para se tornar um serviço para toda a comunidade de Computação.

O Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC) é o principal evento da Comissão Especial de Redes e Sistemas Distribuídos (CE-RES-D) e é organizado em conjunto com o Laboratório Nacional de Redes de Computadores (LARC). A definição das instituições que organizam o SBRC e das cidades que abrigarão o evento é feita com dois ou três anos de antecedência. Foi durante o SBRC 2001, em Florianópolis, que o Prof. Lisandro Zambenedetti Granville e a Profa. Maria Janilce Bosquioli Almeida (UFRGS) receberam a

confirmação de que o SBRC 2004 estava agendado para ser realizado em Gramado. No ano seguinte, no SBRC 2002 de Búzios, essa informação foi ratificada e uma primeira apresentação do SBRC 2004 foi realizada à plateia.

A gênese do JEMS remonta à necessidade de localizar e adaptar ferramentas globais à realidade brasileira. O sistema surgiu como uma versão adaptada — e, mais tarde, profundamente modificada — do EDAS [5]. Ao longo dos anos, o JEMS distanciou-se de sua origem para dar suporte aos fluxos de trabalho específicos da SBC, tornando-se uma solução genuinamente nacional.

2.2 **Das Versões Iniciais ao JEMS3**

A trajetória do JEMS pode ser compreendida em três grandes fases. O **JEMS1**, primeira versão oficial adotada pela SBC a partir de 2004, estabeleceu os fundamentos do sistema: submissão de artigos, gestão de revisores e notificações por e-mail. Funcional e relativamente simples, ele atendeu bem às demandas de uma comunidade ainda em crescimento. O **JEMS2** representou uma expansão significativa: novas funcionalidades foram adicionadas ao longo dos anos- suporte a subventos, formulários de revisão configuráveis, exportação de citações- num modelo de desenvolvimento incremental que, com o tempo, gerou uma base de código de difícil manutenção e com limitações arquiteturais crescentes.

A transição para o **JEMS3** foi motivada precisamente pela necessidade de superar essas limitações [9]. Um dos pontos mais críticos nesse processo foi a decisão estratégica de não sincronizar, inicialmente, a nova base de dados com as versões anteriores [6]. Essa escolha permitiu que o time de desenvol-

2 Histórico e Evolução do JEMS

vimento redesenhasse o esquema do banco de dados sem os limites da compatibilidade retroativa, viabilizando a implementação de modelos mais robustos e flexíveis. Para garantir o acesso dos usuários já existentes, foram importadas todas as contas mantidas na base anterior- aproximadamente 200 mil- para a nova plataforma. Essa migração ocorreu em 2020. O JEMS3 é hoje a plataforma oficial da SBC, enquanto as versões anteriores permanecem disponíveis exclusivamente para acesso a dados legados.

A Tabela 2.1 resume os principais marcos dessa evolução.

Tabela 2.1: Principais marcos da evolução do JEMS

Marco	Descrição	Impacto Arquitetural
Adaptação Inicial (EDAS)	Download, implantação e modificações do sistema no contexto da SBC	Criação da base funcional para gestão de revisões
Consolidação (JEMS1/2)	Expansão de funcionalidades e integração institucional	Estabelecimento do JEMS como padrão na SBC
Reengenharia (JEMS3)	Mudança para infraestrutura moderna e novo esquema de dados	Melhoria na escalabilidade e preparação para integrações

Cada um desses marcos representa não apenas uma mudança técnica, mas também uma resposta às demandas crescentes da

2 Histórico e Evolução do JEMS

comunidade científica. A fase inicial estabeleceu a viabilidade do sistema; a consolidação nas versões JEMS1 e JEMS2 demonstrou sua capacidade de sustentar o ecossistema de eventos em escala nacional; e a reengenharia do JEMS3 representou uma aposta deliberada na longevidade e na modernidade arquitetural, priorizando escalabilidade e abertura a futuras integrações.

O padrão de evolução adotado- substituição gradual em vez de ruptura abrupta- é característico de sistemas que conseguem sobreviver por décadas. Em nenhum momento da transição entre versões houve interrupção de serviços: os eventos em andamento foram mantidos nas versões legadas enquanto o JEMS3 era desenvolvido e estabilizado. Esse cuidado reflete a responsabilidade que a SBC e a equipe de desenvolvimento assumiram ao tornar o JEMS parte essencial do calendário científico nacional.

A história do JEMS é, nesse sentido, inseparável da história da própria SBC. À medida que a comunidade cresceu- em número de eventos, de pesquisadores e de publicações-, o sistema cresceu junto. Os capítulos seguintes descrevem como esse crescimento se traduziu em escolhas arquiteturais concretas, em funcionalidades cuidadosamente projetadas e em integrações que ampliam o alcance e o impacto do sistema para além da gestão de eventos.

3 O JEMS3: Arquitetura e Pilha Tecnológica

O JEMS3 não é apenas uma atualização do sistema anterior- é o resultado de uma reengenharia profunda, motivada pela necessidade de superar limitações arquiteturais acumuladas ao longo de anos de crescimento orgânico. Este capítulo descreve as escolhas tecnológicas que fundamentam o JEMS3, a metodologia de desenvolvimento adotada pela equipe e a linha do tempo de sua construção, desde 2018 até os dias atuais.

3.1 Fundamentos Arquiteturais

A arquitetura do JEMS3 é construída sobre os *frameworks* **Django** e **Angular**, uma escolha que prioriza a agilidade no desenvolvimento e a aderência a padrões estabelecidos de engenharia de software [7]. O sistema adota o padrão *Model-View-Controller* (MVC) para separar a lógica de negócio, a camada de apresentação e a camada de gestão de dados [4]. Essa separação é vital em um sistema no qual as regras de negócio- como a gestão de conflitos de interesse entre autores e revisores- são complexas e exigem avaliações rigorosas em múltiplos níveis.

3.2 Metodologia de Desenvolvimento e Garantia de Qualidade

O desenvolvimento do JEMS3 adota práticas modernas de engenharia, incluindo o uso de *solution designers* e metodologias ágeis para responder rapidamente às demandas da SBC [3]. A garantia de qualidade é implementada por meio de ambientes de homologação (*staging*) rigorosos, nos quais as funcionalidades são testadas antes de serem promovidas à produção. A segurança é tratada como um requisito transversal, abordando proativamente vulnerabilidades como falhas de autenticação e exposição de dados sensíveis.

A containerização com Docker padronizou o ambiente de execução entre os servidores de desenvolvimento, homologação e produção, simplificando os processos de integração e entrega contínuas (CI/CD). A garantia de qualidade também incorporou artefatos multimídia- capturas de vídeo e *screenshots*- para o rastreamento de defeitos de interface, e testes de compatibilidade entre navegadores (Google Chrome e Apple Safari).

3.3 Linha do Tempo do Desenvolvimento

O desenvolvimento do JEMS3 teve início em 2018 e evoluiu de forma consistente ao longo dos anos subsequentes.

2018. O projeto iniciou-se com a estruturação da interface de usuário, abordando configurações de *layout*, exibição de botões em dispositivos móveis e discussões sobre a adoção do modelo *Progressive Web App* (PWA).

3 O JEMS3: Arquitetura e Pilha Tecnológica

2019. A equipe focou no fluxo de submissão e gerenciamento de perfis. Foram construídas APIs de *backend* para trilhas e arquivos de submissão, e iniciado o desenvolvimento da funcionalidade de ordenação de artigos (*Ranking*) com base em médias de pontuações de revisão.

2020. O sistema passou a registrar etapas de submissão com os *status*: *Active*, *Pending*, *Withdrawn*, *Accepted*, *Rejected* e *Published*. Foram implementados o sistema de sugestão de atribuição de revisores (*Assign Suggestion*), alertas de conflito de interesse, controles de prazo e notificação em lote aos autores.

2021. A equipe desenvolveu a funcionalidade de *Rebuttal*, um subsistema de gerenciamento de *templates* de e-mails, suporte a subventos e uma *Wizard* para auxiliar coordenadores. Foi criada também uma *Wiki* com a versão inicial do manual do sistema.

2022. As alterações tiveram foco em desempenho e segurança. O mecanismo de verificação de permissões foi migrado do *frontend* para o *backend*, foram estruturados métodos de paginação e consultas SQL otimizadas, e ocorreu a migração de servidores do ambiente de produção.

2023. Foram implementados processamento em lotes de artigos e e-mails, suporte a fusos horários nos formulários, mecanismos de *log* e auditoria, relatórios estatísticos de submissões e a primeira versão da funcionalidade de criação automática de anais.

2024. Houve aprimoramento na estrutura de permissões em subventos, com transferência de papéis de administração em

3 O JEMS3: Arquitetura e Pilha Tecnológica

casca. Foi desenvolvida a primeira versão do subsistema de integração com a SOL.

2025. A equipe desenvolveu integração com serviços de parceiros institucionais, com destaque para *callbacks* de confirmação de transferência de direitos para a IEEE. Foi implementada uma ferramenta de suporte em tempo real via Zammad, e realizadas reestruturações de APIs para reduzir o tráfego de dados no cliente.

A linha do tempo acima revela um desenvolvimento orientado pelas necessidades reais da comunidade: as otimizações de 2022 responderam a problemas de desempenho identificados em eventos de grande porte; o suporte a fusos horários de 2023 respondeu a demandas de eventos com participação internacional; e a integração com a SOL de 2024 respondeu à necessidade de reduzir o trabalho manual de editoração. Cada iteração foi informada pelo uso real do sistema- o que explica, em grande parte, a sua longevidade e relevância.

A arquitetura modular do JEMS3 também permite que inovações tecnológicas emergentes- como inteligência artificial para atribuição de revisores ou mecanismos de *blockchain* para garantia de integridade de registros [1]- sejam incorporadas de forma gradual, sem comprometer a estabilidade em produção. É sobre as funcionalidades concretas que essa arquitetura viabiliza que trata o próximo capítulo.

4 Perfis de Usuários e Fluxos de Trabalho

A complexidade do JEMS3 reside na diversidade de seus usuários e na especificidade de seus papéis. O sistema mapeou meticulosamente os fluxos de trabalho para cinco perfis principais, garantindo que as permissões e a visibilidade dos dados respeitem a integridade do processo de revisão por pares. A Tabela 4.1 apresenta uma visão geral desses perfis e suas atribuições.

A separação entre esses perfis é implementada por meio de controle de acesso granular (ACL), que governa a renderização de cada elemento da interface conforme o papel do usuário no contexto de cada evento. Um mesmo usuário pode ter permissões distintas em eventos diferentes- e a visibilidade de informações sensíveis, como a identidade dos autores em processos de revisão cega, é garantida sistematicamente em todos os níveis.

O **Autor** é guiado por etapas que incluem o cadastro de todos os coautores- um requisito crítico para evitar fraudes acadêmicas-, a inserção de metadados e o *upload* de arquivos anonimizados. O sistema bloqueia a exclusão de arquivos após o encerramento do prazo, garantindo a integridade do material submetido.

O **Revisor** conta com uma interface para indicação de tópicos de interesse e declaração de conflitos de interesse. Durante a fase de *Bidding*, sinaliza ativamente os trabalhos que deseja

4 Perfis de Usuários e Fluxos de Trabalho

Tabela 4.1: Perfis de usuários do JEMS3

Perfil	Principais ferramentas	Acesso especial
Autor	Submissão de artigos, cadastro de coautores, upload de arquivos anonimizados	Bloqueado após encerramento do prazo
Revisor	Declaração de tópicos de interesse, conflitos de interesse, formulário de avaliação	Bidding, Double Blind
Chair	Criação de trilhas, atribuição de revisores, dashboards analíticos	Visibilidade de autores, estatísticas
Publication Chair	Exportação camera-ready, integração com a SOL	Arquivos finais aceitos
Session Chair	Acesso a resumos e biografias dos autores	Apenas leitura

avaliar, visualizando resumos sempre com o bloqueio de autoria do modo *Double Blind*. Esse insumo alimenta diretamente o algoritmo de atribuição automática de revisores.

O **Chair** possui ferramentas de gestão de alto nível: criação de trilhas, atribuição automática ou manual de revisores e *dashboards* analíticos com submissões por *status* e quantidade de revisões completadas. O **Publication Chair** tem acesso privilegiado aos arquivos finais (*camera-ready*) e às ferramentas de exportação para a SOL. O **Session Chair** dispõe de acesso simplificado aos resumos e biografias dos autores para facilitar a condução das sessões.

Além dos fluxos regulares, o JEMS3 trata com cuidado os caminhos excepcionais. O sistema implementa uma fase de *Rebuttal*, na qual autores respondem às avaliações por meio da mediação dos coordenadores, mantendo o anonimato quando necessário. Na delegação de revisões, caso um revisor substituto recuse a tarefa, o sistema reverte automaticamente a submissão ao avaliador original e dispara as notificações cabíveis. Essa atenção aos fluxos excepcionais- e não apenas aos caminhos felizes- é uma das marcas distintivas da maturidade do JEMS3 como plataforma de gestão científica.

Por fim, o suporte à representação de usuários (*User Proxy*) permite que administradores assumam temporariamente a identidade de outro usuário para fins de suporte técnico, com registros detalhados de auditoria que garantem a rastreabilidade de todas as ações realizadas em modo de representação. A combinação de perfis bem definidos, controle de acesso granular e mecanismos robustos para situações excepcionais faz do JEMS3 um sistema preparado para atender às demandas mais exigentes da gestão de eventos científicos de qualquer porte.

5 Principais Funcionalidades

O JEMS3 é o resultado de anos de desenvolvimento contínuo, guiado pelas necessidades reais da comunidade científica brasileira. Suas funcionalidades cobrem todo o ciclo de vida de um evento científico- da criação e configuração até a publicação dos anais- e podem ser organizadas nos grupos temáticos descritos a seguir.

5.1 Gestão de Eventos e Submissões

O sistema oferece interfaces administrativas para gerenciar todo o ciclo de vida de um evento, desde a solicitação de criação até a publicação dos anais. Foram implementadas funcionalidades de *Event Request*, configuração de identificadores de URL sincronizados com o acrônimo do evento e relações hierárquicas entre eventos maiores e seus subventos (*Eventos Pais* e *Eventos Filhos*). O fluxo de submissão é controlado por meio das *Trackfiles*, que permitem aos coordenadores estipular janelas de envio, restrições de formato, limite de páginas e tamanho de arquivo.

5.2 Avaliação por Pares e Gestão do Comitê

A plataforma implementa rigorosos mecanismos de avaliação cega (*Double Blind*), garantindo que os dados de autoria sejam omitidos nas visões dos revisores e membros do Comitê de Programa (TPC), permanecendo visíveis apenas para os *Chairs*. O sistema conta com múltiplas vias de atribuição de revisores: manual, automática- baseada em algoritmos de *bids* e tópicos de interesse- e externa. A funcionalidade de *Bidding* permite que membros do TPC sinalizem os trabalhos que desejam avaliar, alimentando diretamente o algoritmo de sugestão de atribuição. O sistema oferece também mecanismos de delegação de revisões com máquina de estados, *logs* de auditoria e reversão automática ao avaliador original em caso de recusa.

5.3 Formulários Dinâmicos e Dashboards Analíticos

O JEMS3 adota um sistema dinâmico de formulários de revisão e submissão, com suporte a perguntas dissertativas, de múltipla escolha e pontuações numéricas, além de ordenação por *drag and drop*. Os *dashboards* analíticos exibem submissões por *status*, quantidade de revisões completadas e cobertura de especialidades do comitê, por meio de gráficos e tabelas com acesso otimizado à API, conforme ilustrado na Figura 5.1.

5 Principais Funcionalidades

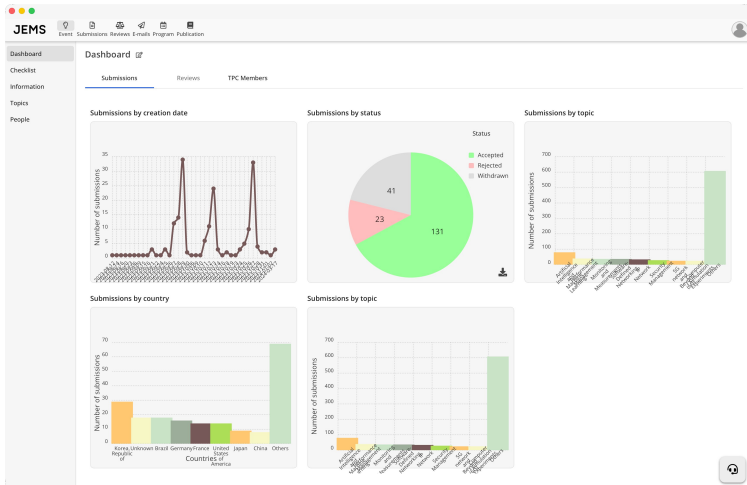


Figura 5.1: Dashboard analítico do coordenador de evento no JEMS3, exibindo submissões por data de criação, por status, por tópico e por país de origem.

5.4 Comunicação e Notificações

O módulo de e-mails utiliza *templates* com substituição de variáveis e envio em fila (assíncrono) para o disparo em lote. Uma interface de *preview* permite que coordenadores visualizem a compilação final do texto- com as variáveis aplicadas- antes do envio, reduzindo o risco de comunicações com erros de formatação.

5.5 Controle de Publicação e Integridade dos Dados

O módulo *Publication Control* introduz o *status* PUBLISHED para as submissões- impedindo retiradas ou edições tardias- e oferece ferramentas de exportação em lote para a compilação dos anais. Uma vez publicada, a submissão é congelada: nenhuma ação destrutiva é permitida, e o *Audit Trail* registra uma linha do tempo auditável de todas as mudanças de estado de cada artigo. Uma *Wizard* interativa auxilia os coordenadores do início da configuração do evento até a publicação final.

5.6 Interface, Acessibilidade e Desempenho

A interface passou por sucessivas modernizações com base nos padrões do *Material Design*, com suporte a *Dark Mode*, internacionalização (i18n) em português e inglês, e responsividade para diferentes tamanhos de tela. O desempenho é garantido por paginação *lazy-loading*, consultas SQL otimizadas e integração com *cache* em Redis. O sistema trata cuidadosamente o suporte a fusos horários- harmonizando bibliotecas conflitantes para refletir os horários locais de cada pesquisador ou o fuso oficial do evento. Mecanismos de dupla confirmação protegem operações irreversíveis, como a exclusão de trilhas com submissões ou o encerramento de fases de revisão.

As funcionalidades descritas neste capítulo formam um ecossistema coerente: a fase de *Bidding* alimenta a atribuição de revisores; os resultados das revisões alimentam o *Rebuttal*; as submissões aceitas alimentam o módulo de publicação e a in-

5 Principais Funcionalidades

tegração com a SOL. Essa cadeia de dependências- cuidadosamente modelada no design do sistema- é o que transforma um conjunto de funcionalidades individuais em uma plataforma coesa para a gestão de eventos científicos, na qual toda ação é rastreável do envio inicial até a publicação permanente.

Cada grupo de funcionalidades foi desenvolvido em resposta a necessidades concretas identificadas pelos usuários ao longo dos anos- o que confere ao JEMS3 uma aderência à prática que dificilmente seria alcançada por um sistema projetado de forma abstrata. É essa conexão entre desenvolvimento e uso que faz do JEMS3 não apenas um sistema funcional, mas uma ferramenta genuinamente útil para a comunidade que o sustenta e que, ao mesmo tempo, o continua moldando. Para uma descrição detalhada de cada uma das 40 frentes evolutivas de desenvolvimento do sistema, consulte o Anexo A.

O conjunto dessas funcionalidades posiciona o JEMS3 como uma plataforma madura, capaz de atender tanto ao pequeno workshop regional quanto ao grande simpósio nacional com centenas de submissões. A amplitude do sistema- que vai da configuração inicial do evento à publicação indexada dos anais- é o resultado direto de mais de duas décadas de escuta ativa da comunidade científica brasileira de Computação.

6 A Integração JEMS3–SOL

Um dos avanços mais relevantes do desenvolvimento recente do JEMS3 é a criação de uma funcionalidade para a exportação automatizada de artigos para a biblioteca digital da SBC, a **SBC OpenLib (SOL)** [10]. Este projeto, originado de uma tese de graduação orientada pelo Prof. Lisandro Zambenedetti Granville na UFRGS, abordou um problema histórico: a importação manual de artigos para a SOL, um processo trabalhoso e propenso a erros. A automação desse processo representa um ganho significativo não apenas em eficiência, mas também em confiabilidade, pois elimina a etapa de transcrição manual de dados- uma das principais fontes de inconsistências nos metadados publicados na biblioteca digital.

Anteriormente, o processo manual exigia que os coordenadores baixassem artigos um a um e transcrevessem os dados para quatro planilhas distintas- artigos, autores, referências e seções-, que eram então convertidas em um arquivo XML submetido à biblioteca. O desenvolvimento da nova funcionalidade no JEMS3 automatiza a extração desses dados diretamente do esquema SQL do sistema, aplicando regras de validação que impedem a exportação de dados incompletos ou mal formados [10]. Isso não apenas economiza centenas de horas de trabalho voluntário dos pesquisadores, mas também garante que os metadados presentes na biblioteca digital sejam idênticos aos aprovados durante o processo de revisão no JEMS3.

A integração foi concebida para ser resiliente a variações nos formatos de conferência e foca na padronização de quatro con-

juntos de dados fundamentais, descritos na Tabela 6.1.

Tabela 6.1: Entidades de dados na integração JEMS3–SOL

Entidade	Atributos Capturados	Objetivo
Metadados de Artigo	Título, Resumo, Idioma, DOI	Garantir indexação em bases internacionais
Cadastro de Autores	Nome Completo, E-mail, Instituição, ORCID	Evitar ambiguidade de autoria e facilitar a busca
Gestão de Referências	Lista de Citações, Formatação ABNT/SBC	Automatizar o grafo de citações da SOL
Estrutura de Evento	Trilhas, Sessões, Datas de Publicação	Organizar os anais de forma hierárquica

O processo de integração envolve cinco etapas principais, descritas na Tabela 6.2. Vale notar que cada etapa foi projetada para ser resiliente a falhas: se um conjunto de dados está incompleto ou mal formado, o sistema impede a progressão para a etapa seguinte, garantindo que apenas metadados íntegros sejam publicados na biblioteca digital.

Na etapa de **extração**, o JEMS3 coleta os metadados das submissões aceitas por meio de consultas SQL otimizadas. Na **transformação**, os dados são organizados nas quatro planilhas padrão da SOL. Na **validação**, uma camada de verificação de

Tabela 6.2: Etapas do processo de integração JEMS3–SOL

Etapas	Ação pelo JEMS3	Mecanismo
Extração	Coleta de metadados das submissões aceitas	Consultas SQL otimizadas
Trans- formação	Organização dos dados nas 4 planilhas padrão da SOL	Módulos de processamento
Validação	Verificação de integridade (DOI, nomes, e-mails)	Camada de validação de modelo
Geração de XML	Conversão das planilhas para o esquema OJS/XML	Conversor integrado ao sistema
Ingestão	Submissão final à biblioteca SOL	Upload via interface administrativa da SOL

integridade confere DOIs, nomes e e-mails [8]. Na **geração do XML**, os dados são convertidos para o esquema OJS/XML compatível com a plataforma que sustenta a SOL [2]. Por fim, na **ingestão**, o arquivo gerado é submetido à biblioteca por meio da interface administrativa da SOL.

A sequência dessas cinco etapas foi projetada para ser ao mesmo tempo rigorosa e transparente. Os coordenadores de eventos acompanham o progresso de cada etapa por meio de

uma interface dedicada no JEMS3, que exibe o status do processo e sinaliza eventuais erros de validação de forma clara e acionável. Esse nível de visibilidade é fundamental para que os responsáveis pela publicação possam intervir rapidamente caso algum dado esteja incompleto ou fora do padrão esperado—sem precisar aguardar a detecção do problema pela equipe da SOL após a submissão do arquivo XML.

Este desenvolvimento não se limitou ao código do JEMS3, mas incluiu também a modificação dos *templates* de artigos da SBC para assegurar que as informações capturadas no momento da submissão estivessem em conformidade com as necessidades da automação. O impacto prático dessa integração é expressivo: coordenadores que anteriormente dedicavam horas à transcrição manual de metadados passaram a contar com um processo automatizado, confiável e auditável. Além disso, a padronização dos metadados garante que os artigos publicados na SOL sejam indexados corretamente em bases internacionais como o Google Scholar e o OpenAlex, aumentando a visibilidade e o impacto da produção científica da comunidade de Computação brasileira.

Do ponto de vista técnico, um dos principais desafios da integração foi lidar com a heterogeneidade dos dados provenientes de diferentes eventos. Cada conferência tem suas próprias convenções de formatação de nomes de autores, afiliações institucionais e referências bibliográficas. O módulo de integração precisou incorporar regras de normalização robustas para harmonizar essas diferenças antes de gerar o XML final, garantindo que o resultado seja sempre compatível com o esquema esperado pela SOL. Esse trabalho de normalização, embora invisível ao usuário final, é fundamental para a qualidade dos metadados disponibilizados na biblioteca digital e, por consequência, para a correta indexação dos artigos nas bases de dados científicas

6 A Integração JEMS3–SOL

internacionais.

A integração JEMS3–SOL representa, em síntese, uma mudança de paradigma: o JEMS3 deixa de ser apenas um sistema de gestão de eventos e passa a ser um elo ativo na cadeia de publicação científica aberta no Brasil. Ao automatizar o caminho entre o aceite de um artigo e sua disponibilização indexada na SOL, o sistema garante que a produção científica da comunidade brasileira de Computação chegue ao mundo com a qualidade, a consistência e a rastreabilidade que ela merece.

É importante reconhecer, por fim, que a integração não é um produto acabado, mas um processo em evolução. À medida que novos padrões de metadados emergem e novas bases de indexação impõem novos requisitos, o módulo precisará ser atualizado- e a arquitetura modular do JEMS3 foi projetada exatamente para acomodar essas evoluções de forma gradual e segura. Esse compromisso com a evolução contínua é parte do que torna o projeto sustentável a longo prazo e é o que garante que a ciência produzida no Brasil continue sendo visível, acessível e impactante no cenário internacional.

7 Considerações Finais

Este livreto narrou a trajetória de um sistema que nasceu como resposta a um problema concreto- organizar as submissões do SBRC 2003- e que, duas décadas depois, sustenta mais de 5 mil eventos, mais de 650 mil revisões e mais de 300 mil autores cadastrados. Essa escala não foi planejada de antemão: ela é o resultado de uma sequência de decisões acertadas, tomadas por uma equipe comprometida com as necessidades reais da comunidade, ao longo de mais de vinte anos de desenvolvimento contínuo.

O percurso do JEMS ilustra um princípio que vale para além do desenvolvimento de software: sistemas duradouros não são os mais ambiciosos, mas os mais responsivos. O JEMS1 não tinha todas as funcionalidades do JEMS3. O JEMS2 acumulou limitações que eventualmente exigiram uma reengenharia profunda. O JEMS3, por sua vez, foi construído com a consciência dessas limitações- optando por uma arquitetura modular, por práticas modernas de DevOps e por uma metodologia de desenvolvimento que prioriza a estabilidade do serviço acima de qualquer inovação pontual. É essa consciência histórica que distingue o JEMS3 de um simples sistema novo: ele é o resultado de mais de duas décadas de aprendizado acumulado.

A integração com a SOL, descrita no Capítulo 6, merece destaque especial nas considerações finais porque representa uma mudança qualitativa no papel do JEMS3 dentro do ecossistema científico da SBC. Antes da integração, o ciclo de vida de um artigo científico era interrompido no momento do aceite: a trans-

7 Considerações Finais

ferência de metadados para a biblioteca digital dependia de intervenção manual, sujeita a erros e atrasos. Com a automação desse processo, o JEMS3 passou a acompanhar o artigo do início ao fim- da submissão à indexação permanente na SOL- tornando-se não apenas um sistema de gestão de eventos, mas um elo essencial da cadeia de publicação científica aberta no Brasil.

Olhando para o futuro, os caminhos mais promissores para o JEMS3 são aqueles que aprofundam o que ele já faz bem, em vez de ampliar seu escopo indiscriminadamente. O aprimoramento dos algoritmos de atribuição de revisores com base em afinidade temática, o suporte a novos modelos de avaliação aberta (*open review*), e o aprofundamento da interoperabilidade com sistemas parceiros- como a IEEE e outras sociedades científicas internacionais- são direções naturais para um sistema que já demonstrou maturidade técnica para integrações complexas. A arquitetura modular adotada no JEMS3 foi projetada exatamente para acomodar essas evoluções de forma gradual e segura. O desafio não será tecnológico- a base está consolidada-; será, como sempre foi, o de manter a escuta ativa da comunidade e a coragem de priorizar o que realmente importa para quem usa o sistema todos os dias.

O JEMS3 é, em essência, um bem público- construído com recursos da universidade, mantido com trabalho voluntário e dedicado à comunidade científica nacional. Essa natureza o distingue de produtos comerciais e lhe confere uma responsabilidade particular: a de permanecer acessível, transparente e orientado ao interesse coletivo, mesmo sob pressões de prazo, de recursos e de escala. É essa responsabilidade que a equipe de desenvolvimento abraçou ao longo de mais de duas décadas- e que continuará orientando o JEMS nas versões que ainda estão por vir.

7 Considerações Finais

Por fim, este livreto não estaria completo sem um reconhecimento explícito das pessoas que tornaram o JEMS3 possível. Os nomes de cada desenvolvedor que contribuiu para a construção e a evolução do JEMS ao longo dos anos estão registrados no Anexo B, como reconhecimento permanente de uma contribuição que, embora muitas vezes invisível ao usuário final, foi essencial para que o sistema chegasse até aqui.

A história do JEMS é, em última análise, a história de uma comunidade que decidiu investir em sua própria infraestrutura- e que colhe, décadas depois, os frutos dessa decisão. Que o JEMS3 continue sendo digno dessa confiança.

Encerramos com uma observação que talvez pareça simples, mas que resume bem o espírito deste projeto: o JEMS3 funciona. Funciona na manhã do prazo de submissão do SBRC, quando centenas de autores enviam seus artigos simultaneamente. Funciona quando um coordenador precisa atribuir 300 revisões em poucos minutos. Funciona quando um artigo aceito precisa chegar à SOL com todos os seus metadados íntegros. Fazer um sistema funcionar nessas condições, de forma consistente, por mais de duas décadas, é uma conquista que merece ser celebrada- e é exatamente isso que este livreto pretende fazer. É a cada uma das pessoas que tornaram isso possível- e cujos nomes estão registrados no Anexo B- que este livreto é dedicado.

Referências Bibliográficas

- [1] 3rd IFIP/IEEE International Workshop on Fully-Flexible Internet Architectures and Protocols for the Next-Generation Tactile Internet (FlexNGIA 2023). <https://www.flexngia.net/workshop-flexngia2023>. Miami, FL, USA, 8 maio 2023. Acessado em abril de 2026.
- [2] Artigos – ERCEMAPI – PETComp. <https://petcompufma.org/ercemapi/artigos.php>. Acessado em abril de 2026.
- [3] Django & Angular – Release Notes. https://guides.rubyonrails.org/3_0_release_notes.html. Acessado em abril de 2026.
- [4] ETC – Encontro de Teoria da Computação – CSBC 2024. <https://csbc.sbc.org.br/2024/etc/>. Acessado em abril de 2026.
- [5] JEMS: Welcome (not logged in). <https://jems.sbc.org.br/>. Acessado em abril de 2026.
- [6] Login - JEMS. <https://jems.sbc.org.br/jems2/index.php?r=site/login>. Acessado em abril de 2026.
- [7] Manual do JEMS (Guest). <https://submissoes.sbc.org.br/manual/index.php>. Acessado em abril de 2026.
- [8] SF – Simpósio Brasileiro de Cibersegurança 2025 (SB-Seg 2025). <https://sbseg2025.ppgia.pucpr.br/en/>

Referências Bibliográficas

- call-for-papers/sf/. PUCPR. Acessado em abril de 2026.
- [9] Tutorial de submissão de artigos - INCITEL 2025. <https://inatel.br/incitel/documents/tutorial-de-submissao-de-artigos-jems3.pdf>. Inatel. Acessado em abril de 2026.
- [10] Raphael Malinski Vieira, Lisandro Zambenedetti Granville, and Marcus Vinicius Brito da Silva. Solução para integração do JEMS3 à SOL: exportando artigos científicos de eventos para a biblioteca digital da SBC. <https://lume.ufrgs.br/bitstream/handle/10183/287611/001241589.pdf>. Lume UFRGS. Acessado em abril de 2026.

A Lista de Funcionalidades do JEMS3

A seguir apresenta-se a lista não exaustiva das 40 frentes evolutivas de desenvolvimento do JEMS3, organizadas em ordem de implementação ao longo do ciclo de desenvolvimento do sistema.

1. **Gestão Central de Eventos.** Interfaces administrativas para gerenciar todo o ciclo de vida do evento, desde a solicitação de criação (*Event Request*) até a publicação dos anais. Inclui configuração de identificadores de URL sincronizados com o acrônimo do evento e o estabelecimento de relações entre *Eventos Pais* e *Eventos Filhos* (subconferências e trilhas).
2. **Fluxo de Submissões e Controle de Arquivos (Trackfiles).** Suporte a regras complexas de envio de manuscritos por meio das *Trackfiles*: janelas de envio, restrições de extensão de arquivo (MIME types), limite de páginas e tamanho máximo. Inclui bloqueio de exclusão de arquivos pelos autores após o encerramento do prazo.
3. **Construtor de Formulários Dinâmicos.** Sistema dinâmico de formulários de revisão e submissão com suporte a perguntas dissertativas, de múltipla escolha e pontuações numéricas (*scores*), com ordenação por *drag and drop*.

4. **Avaliação Cega (Double Blind).** Modo *Double Blind* que garante a omissão completa do campo de autores e filiações nas visões dos revisores e membros do Comitê de Programa (TPC), permanecendo visíveis apenas para os *Chairs*.
5. **Gestão do Comitê de Programa (TPC) e Convites.** Fluxo de convites por e-mail para formação do corpo editorial, com seleção de tópicos de interesse e rejeição, limitados por parâmetros de mínimo e máximo definidos pelo evento.
6. **Atribuição de Revisores e Conflitos de Interesse.** Múltiplas vias de atribuição- manual, automática (baseada em algoritmos de *bids* e tópicos de interesse) e externa-, com validações de conflito de interesse, mascaramento de e-mails em certas telas e bloqueio de delegações indevidas.
7. **Motor de Comunicação e E-mails (Fila e Templates).** Módulo de e-mails com *templates* de substituição de variáveis e envio em fila (assíncrono) para disparo em lote, evitando *timeouts* e travamento da interface.
8. **Representação de Usuário (User Proxy) e Segurança.** Funcionalidade de *Proxy* que permite a um Admin/*Chair* assumir a identidade de outro usuário para fins de suporte técnico, com registro detalhado de *logs* de auditoria.
9. **Dashboards Analíticos e Estatísticas.** Gráficos e tabelas exibindo submissões por *status* (Pending, Active, Accepted, Rejected), quantidade de revisões completadas e cobertura de especialidades do comitê, com dados particionados para reduzir o peso das requisições à API.

10. **Organização do Programa e Sessões Técnicas.** Módulo de programa do evento com granularidade temporal, associação de submissões a blocos de horário, designação de *Session Chairs* e diálogos personalizados para gestão do calendário.
11. **Manipulação Complexa de Datas e Fusos Horários.** Esforço estrutural de unificação do tratamento de datas, harmonizando bibliotecas conflitantes (Moment.js, Day.js, Date nativo) com suporte extensivo a *timezones* para refletir os horários locais de cada pesquisador ou o fuso oficial do evento.
12. **Controle de Publicação e Anais (Proceedings).** Módulo *Publication Control* com o *status* PUBLISHED (impedindo novas edições ou *withdraws*), ferramentas de exportação em lote para compilação dos anais e integração de metadados obrigatórios (ex.: formulários de direitos autorais IEEE *Copyright*).
13. **Interface de Usuário (UI) e Responsividade.** Modernizações baseadas nos padrões do *Material Design*, com refatorações de tabelas e menus de navegação para dispositivos móveis, e atualizações do *framework* Angular (incluindo migração para Angular 12 com otimizações de compilação Ivy).
14. **Acessibilidade, i18n e Tema Escuro (Dark Mode).** Suporte à internacionalização (i18n) com tradução de interfaces, *checklists* e variáveis para pt-BR e inglês, além de suporte nativo ao *Dark Mode* com correções iterativas de bugs visuais de contraste.
15. **Desempenho, Bancos de Dados e Caching (Redis).** Criação de consultas SQL diretas para listagem de submissões,

paginação *lazy-loading* no *frontend* e integração com *cache* em Redis, com alertas na interface sobre a propagação de alterações.

16. **Fluxo de Delegação e Repasse de Revisões.** Máquina de estados complexa para delegação de revisões, com *endpoint* dedicado, *logs* de auditoria precisos e reversão automática ao avaliador original em caso de recusa do substituto.
17. **Arquitetura de Build, Compilação e Motor Ivy.** Investigações e ajustes sobre a compilação do Angular para produção (`ng build -prod`), resolvendo incompatibilidades de módulos de terceiros decorrentes da ativação do motor Ivy.
18. **Gestão de Estado e Ciclo de Vida de Navegação (Routing).** Correção de vulnerabilidades no armazenamento de estado entre telas, eliminando erros de referência nula em caminhos não convencionais e resolvendo problemas de ciclo de vida que causavam duplicação de dados em listas.
19. **Refatoração Estrutural e Componentização Compartilhada.** Componentização rigorosa de telas pesadas- separando, por exemplo, os componentes de *Submissão* e *Revisões* em elementos modulares compartilhados- para prevenir renderizações desnecessárias e isolar contextos de dados.
20. **Mecanismos de Filtros Globais e Buscas Diretas.** Correção de filtros essenciais (como *Show old events too*) e otimização da usabilidade de busca com inclusão de pesquisa direta por ID do artigo.

21. **Controle de Autorização e Resolução de Permissões (ACL).** Refatorações para vincular corretamente as credenciais do usuário às permissões do escopo do evento (*ACL-Access Control List*), corrigindo erros 403 Forbidden e garantindo que apenas usuários validados alterem o estado do banco de dados.
22. **Personalização de Quotas e Metas de Revisão (Review Settings).** Configuração dinâmica da cota de revisores por artigo (ex.: 4 revisões para um *paper* específico em vez do padrão de 3), com adaptação dos componentes visuais do *Chair* para monitorar a porcentagem de conclusão com base em métricas variáveis.
23. **Transferência e Realocação de Submissões entre Trilhas.** Funcionalidade que permite ao *Chair* mover submissões entre diferentes trilhas (*tracks*), listando dinamicamente apenas os eventos na linha de filiação que o usuário administra, por meio de componentes de diálogo (modais) específicos.
24. **Containerização, Infraestrutura e DevOps (Docker).** Adoção de contêineres para os ambientes de *frontend*, *backend* e infraestrutura, padronizando o ambiente de execução da máquina local do desenvolvedor ao servidor de produção da SBC e facilitando os processos de CI/CD.
25. **Transições Visuais de Estado e Feedbacks de Avaliação.** Interface que distingue claramente os microestados do ciclo de revisão, como a transição visual entre uma revisão *Assigned* (atribuída) e uma revisão *Notified* (notificada), mitigando confusão dos gestores sobre o status da comunicação com o comitê.

26. **Interface de Validação e Preview de Comunicações.** Reformulação do módulo de envio de e-mails em massa com fluxo em duas etapas- *Preview* (visualização com variáveis aplicadas) e *Send*- reduzindo o risco de envio de mensagens com erros de formatação.
27. **UX Preventiva em Ações Críticas.** Introdução da mecânica de *dupla confirmação* em ações destrutivas- como exclusão de sessões ou encerramento de fases de revisão- para proteger a integridade dos dados.
28. **Categorização Visual e Paletas de Cores (Color Pickers).** Integração de módulos de seleção de cores personalizadas (NgxMatColorPickerModule) para atribuição de cores a trilhas, *tags* ou blocos de sessão no cronograma.
29. **Formatação Regional e Controle de Locales.** Resolução do *bug* de *locale* para garantir que calendários e seletores de data/hora do Angular respeitem a formatação regional do usuário (DD/MM/AAAA para brasileiros, formato de 12h/24h conforme região).
30. **Organização de Formulários Densos via Sistema de Abas (Tabs).** Particionamento de formulários longos (como a gestão de sessões) em abas temáticas- separando configurações básicas da interface de vinculação de artigos- para reduzir rolagem excessiva e cansaço visual.
31. **Sistema de Checklist (Wizard) e Guias Interativos.** Componente visual de *checklist* (assistente passo a passo) para processos complexos, como configuração inicial de evento ou controle de publicação, integrado ao sistema de i18n e com preservação de estado entre etapas.

32. **Validação Granular de Uploads e Tolerância a Case Sensitivity.** Proteções no módulo de *Trackfiles* para aceitar variações de nomenclatura de extensões de arquivo (ex.: .PDF e .pdf), evitando rejeições indevidas causadas por diferenças de capitalização.
33. **Histórico de Submissões e Rastreabilidade (Audit Trail).** Funcionalidade de rastreamento do histórico de submissões com linha do tempo auditável das mudanças de estado (Pending, Active, Accepted, Rejected, Published), garantindo a integridade do processo editorial.
34. **Interface de Bidding (Declaração de Interesse em Avaliação).** Interface para o período de *Bidding*, na qual membros do TPC visualizam resumos- sempre respeitando o *Double Blind*- e sinalizam ativamente os trabalhos que desejam avaliar, alimentando o algoritmo de sugestão de atribuição.
35. **Gestão de Identidade Visual e Cache de Avatares.** Refinamentos no gerenciamento de estado do *frontend* para atualizar corretamente as imagens de perfil (*avatars*) ao entrar ou sair do modo *Proxy*, com funções de *refresh* forçado no ciclo de navegação.
36. **Proteção de Integridade Pós-Publicação (Freeze States).** Mecanismos para bloquear completamente ações destrutivas- como *withdraws* de autores e edições tardias em formulários de revisão- uma vez que uma submissão atinge o *status* PUBLISHED.
37. **Hierarquia Estrutural de Eventos (Pais e Filhos).** Modelagem de dados hierárquica para eventos acadêmicos

de grande porte, com interfaces que permitem ao administrador transitar e gerenciar submissões respeitando a linha de filiação entre eventos pais e filhos.

38. **Homologação e Compatibilidade Cross-Browser.** Testes rigorosos de qualidade para garantir funcionamento unificado em diferentes navegadores, com investigações documentadas de inconsistências entre Google Chrome e Apple Safari.
39. **Políticas de Visibilidade Restrita (Role-based Views).** Refatorações para garantir que painéis amplos (como *Show All Papers*) apliquem os mesmos filtros de omissão de dados que as páginas individuais de avaliação, por meio de ACL granular governando a renderização de fragmentos de interface.
40. **Gerenciamento de Artefatos de QA e Depuração Visual.** Adoção de capturas de vídeo (.mp4) e *screenshots* da tela de desenvolvedor- anexados diretamente às requisições do GitHub- para rastreamento de defeitos de animação, transições de roteamento e vazamentos de memória.

B Colaboradores do JEMS

Ao longo de mais de duas décadas, o JEMS foi construído e evoluído pelo esforço coletivo de dezenas de pessoas. Este anexo tem como objetivo reconhecer nominalmente aqueles que contribuíram para o desenvolvimento e a evolução do sistema ao longo dos anos, pesquisadores, engenheiros de software, membros da SBC, estudantes de graduação e pós-graduação, e colaboradores técnicos que, em diferentes momentos e com diferentes papéis, deixaram sua marca no código, na arquitetura e na experiência de uso do JEMS. Ainda que essa lista não seja exaustiva, que os nomes listados abaixo sirvam de registro permanente de uma contribuição que, embora muitas vezes invisível ao usuário final, foi essencial para que o sistema chegasse até aqui.

- Alamir Bobroski Filho
- Ana Pagnoncelli
- Andrei Rodrigues
- Andrey Quintero González
- Annelise Abreu Coutinho Arruda
- Briggette Roman
- Cassio Fachinelli

B Colaboradores do JEMS

- Cleber Machado Ortiz
- Francisco Knebel
- Fábio Fabian Daitx
- Felipe Formiga
- Gabriel Lima
- Geraldo Teixeira do Nascimento Filho
- João Raphael
- João Vitor de Souza
- Juan Marcelo Trabaina
- Leonardo Reis
- Leonardo Rocha dos Santos
- Leonardo Santos
- Lucas Bondan
- Lucas G. Silveira
- Luciano Zembruzki
- Rafael Pereira Esteves
- Raphael Malinski
- Renata Neuland
- Thiago Trautwein Santos
- Tiago Schmidt
- Tobias Soares
- Weverton Luis da Costa Cordeiro



[instagram.com/sbcoficial](https://www.instagram.com/sbcoficial)



[facebook.com/sbcbrasil](https://www.facebook.com/sbcbrasil)



+55 51 3308-6835

[Acesse **www.sbc.org.br**](http://www.sbc.org.br)

