

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
SISTEMAS DE INFORMAÇÃO

Victor dos Santos Camargo

**Desenvolvimento de um Sistema de Preparação de Documentos Eletrônicos para Gestão
de Fluxos de Assinatura Digital**

Florianópolis
2026

Victor dos Santos Camargo

Desenvolvimento de um Sistema de Preparação de Documentos Eletrônicos para Gestão de Fluxos de Assinatura Digital

Trabalho de Conclusão de Curso submetido ao
Curso de Graduação em Sistemas de Informa-
ção do Centro Tecnológico da Universidade
Federal de Santa Catarina como requisito para
obtenção do título de Bacharel em Sistemas de
Informação.

Orientador: Prof. Frederico Schardong, Dr.

Coorientador: Prof. Ricardo Felipe Custódio,
Dr.

Florianópolis

2026

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Camargo, Victor dos Santos

Desenvolvimento de um sistema de preparação de documentos eletrônicos para gestão de fluxos de assinatura digital / Victor dos Santos Camargo ; orientador, Frederico Schardong, coorientador, Ricardo Felipe Custódio, 2026.

85 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Sistemas de Informação, Florianópolis, 2026.

Inclui referências.

1. Sistemas de Informação. 2. Engenharia de software. 3. Verificação de requisitos. 4. Sistema de assinaturas digitais. 5. Gestão de fluxos documentais. I. Schardong, Frederico. II. Custódio, Ricardo Felipe. III. Universidade Federal de Santa Catarina. Graduação em Sistemas de Informação. IV. Título.

Victor dos Santos Camargo

Desenvolvimento de um Sistema de Preparação de Documentos Eletrônicos para Gestão de Fluxos de Assinatura Digital

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de Bacharel em Sistemas de Informação e aprovado em sua forma final pelo curso de Graduação em Sistemas de Informação.

Florianópolis, Junho de 2026.

Prof. José Eduardo de Lucca, Me.
Coordenador do Curso

Banca Examinadora:

Prof. Frederico Schardong, Dr.
Orientador
Instituto Federal do Rio Grande do Sul

Prof. Ricardo Felipe Custódio, Dr.
Coorientador
Universidade Federal de Santa Catarina

Brendon Vicente Rocha Silva
Avaliador

Giulia Manno Lima
Avaliador

Aos meus pais, Arthur e Dulci, pelo suporte incondicional; à
Gabi, pelo amor e paciência; e ao meu gato esfomeado.

AGRADECIMENTOS

Ao Laboratório de Segurança em Computação (LabSEC) e a todos os seus integrantes, pelo ambiente de excelência e cooperação. Agradeço a oportunidade de aprendizado e o incentivo constante, que foram fundamentais não apenas para a realização deste trabalho, mas para o meu amadurecimento acadêmico e profissional. Ao professor Frederico e ao professor Custódio, pela confiança depositada e pelo suporte constante.

RESUMO

Processos antes físicos estão sendo adaptados cada vez mais em formato digital por oferecerem praticidade e eficiência, como é o caso das assinaturas digitais em documentos eletrônicos. Nesse contexto, o projeto ADvanced Electronic Signature (ADES) oferece sistemas de assinatura, verificação e compartilhamento de documentos *Portable Document Format* (PDF) com autenticação do usuário, mas ainda carece de mecanismos mais ricos para gerenciar o trâmite desses arquivos. Este trabalho apresenta o Preparador de Documentos (PdD), concebido como um conjunto integrado de funcionalidades que viabiliza o controle sobre o fluxo documental por meio do cadastro e especificação de múltiplos signatários, estipulação de prazos limite, definição de níveis de comprometimento e automatização de notificações. A abordagem metodológica contempla o levantamento de requisitos junto a gestores do projeto, a modelagem de processos em notação *Business Process Model and Notation* (BPMN), o desenvolvimento e inserção dos novos mecanismos nas arquiteturas de *frontend* (Portal de Assinatura (PAS)) e *backend* (Gerenciador Eletrônico de Documentos (GED)), e a respectiva verificação do sistema. Por fim, descreve-se o atendimento integral do escopo proposto, confrontando cada requisito funcional com o comportamento e as telas da aplicação, amparado por ilustrações de resultados e uma tabela de rastreabilidade.

Palavras-chave: Engenharia de software. Verificação de requisitos. Sistema de assinaturas digitais. Gestão de fluxos documentais.

ABSTRACT

Processes previously handled physically are increasingly being adapted into digital formats due to the practicality and efficiency they offer, as seen in digital signatures for electronic documents. In this context, the ADvanced Electronic Signature (ADES) project provides systems for signing, verifying, and sharing PDF documents with user authentication, yet it still lacks richer mechanisms to manage the workflow of these files. This work introduces the Document Preparer (PdD), conceived as an integrated set of features that enables control over the documentary workflow through the registration and specification of multiple signers, definition of deadlines, establishment of commitment levels, and automated notifications. The methodological approach encompasses requirement gathering with project managers, process modeling using BPMN notation, the development and insertion of these new mechanisms into the frontend (Signature Portal (PAS)) and backend (Electronic Document Manager (GED)) architectures, and the respective verification of the system. Lastly, the complete fulfillment of the proposed scope is described, confronting each functional requirement with the application's behavior and user interfaces, supported by illustrations of the results and a dedicated traceability table.

Keywords: Software engineering. Digital signature system. Document workflow management. Requirement verification.

LISTA DE TABELAS

Tabela 1 – Requisitos Funcionais do Preparador de Documentos.	35
Tabela 2 – Atendimento dos requisitos funcionais pelo fluxo de telas proposto em trabalho anterior.	43
Tabela 3 – Organização das pastas do Preparador de Documentos no código-fonte do Portal de Assinatura.	54
Tabela 4 – Atendimento aos Requisitos Funcionais do Preparador de Documentos. . . .	83

LISTA DE FIGURAS

Figura 1 – Processo de assinatura novo documento no fluxo antigo.	26
Figura 2 – Tela da página principal.	27
Figura 3 – Tela de autenticação.	27
Figura 4 – Tela com documento selecionado para assinatura.	28
Figura 5 – Tela de progressão do processo de assinatura.	28
Figura 6 – Tela de visualização do resultado da assinatura.	29
Figura 7 – Mapa de navegação do fluxo de compartilhamento de documentos e coleta de assinaturas.	30
Figura 8 – Tela ao clicar em compartilhar com documento selecionado e autenticado. .	30
Figura 9 – Tela exemplo de autenticação.	38
Figura 10 – Tela exemplo da lista de documentos.	38
Figura 11 – Tela exemplo de documento que foi preparado.	39
Figura 12 – Tela exemplo de especificação de signatários.	39
Figura 13 – Tela exemplo de adição de signatário no processo de preparo.	40
Figura 14 – Tela exemplo da finalização do processo de preparo.	40
Figura 15 – Tela exemplo da finalização do processo de preparo.	41
Figura 16 – Arquitetura do sistema ADES.	45
Figura 17 – Processo proposto de preparo de documentos e coleta de assinaturas. . . .	47
Figura 18 – Etapas propostas do preparo de um documento no Preparador de Documentos.	47
Figura 19 – Fluxo proposto para selecionar um novo documento.	48
Figura 20 – Fluxo proposto de inclusão de signatário e posicionamento no PDF.	48
Figura 21 – Processo proposto de coleta de assinaturas.	49
Figura 22 – Processo proposto de assinatura pelo signatário esperado a partir do e-mail. .	50
Figura 23 – Processo proposto de recebimento do documento finalizado.	50
Figura 24 – Processo proposto de download do documento a partir do link no <i>site</i>	51
Figura 25 – Fluxo proposto de assinatura com um clique sem posicionamento de artefato visual.	51
Figura 26 – Fluxo proposto de compartilhamento por link sem especificar signatários. .	52
Figura 27 – Processo proposto de compartilhamento por link de documento já inserido no sistema.	53
Figura 28 – Modelagem lógica das relações do banco de dados antes do preparador. . . .	56
Figura 29 – Modelagem lógica das relações do banco de dados proposta para o preparador.	57
Figura 30 – Documento PDF de exemplo utilizado nos testes do fluxo.	61
Figura 31 – Nova tela inicial do Portal de Assinatura.	61
Figura 32 – Lista de documentos do usuário vazia e com botão para novo documento. . .	62
Figura 33 – Seleção do arquivo PDF para novo documento no sistema operacional Ubuntu.	62

Figura 34 – Tela de preparo do documento: estado inicial após selecionar documento novo, com campos e botões para <i>Assinar</i> ou <i>Compartilhar</i> , usados nos fluxos das seções seguintes.	63
Figura 35 – Indicação visual para posicionar o artefato de assinatura sobre o PDF.	64
Figura 36 – Formulário de dados do signatário no preparo do documento.	64
Figura 37 – Formulário de dados do signatário no preparo do documento após clicar em <i>Sou eu</i>	65
Figura 38 – Tela de preparo após o usuário se adicionar como signatário.	65
Figura 39 – Documento com 2 signatários adicionados, com marcadores de assinatura sobre o documento e nome e descrição definidos.	66
Figura 40 – Estado de processamento após o envio do documento.	67
Figura 41 – Confirmação de sucesso após o preparo do documento.	67
Figura 42 – Monitoramento do progresso de coleta de assinaturas de um documento preparado.	68
Figura 43 – Lista de documentos preparados com indicação do estado de completude.	68
Figura 44 – Email de notificação para signatário esperado.	69
Figura 45 – Tela do processo de assinatura para o signatário.	70
Figura 46 – Tela após a conclusão da assinatura pelo signatário esperado.	70
Figura 47 – Visualização do documento e lista de signatários antes ou após assinar.	71
Figura 48 – Formulário de cadastro de signatário com meio de contato (e-mail ou número de telefone para WhatsApp).	72
Figura 49 – Mensagem de pedido de assinatura recebida pelo signatário no WhatsApp.	72
Figura 50 – Email de notificação para quem preparou o documento.	73
Figura 51 – Tela de visualização do documento finalizado.	74
Figura 52 – Lista de documentos com documentos em diversos estados de completude.	75
Figura 53 – Email de notificação para quem preparou o documento.	76
Figura 54 – Tela de visualização do documento finalizado incompleto.	77
Figura 55 – Aviso prévio sobre situação atual do documento incompleto antes de baixar.	77
Figura 56 – Email de confirmação de assinatura concluída para o signatário.	78
Figura 57 – Link de compartilhamento gerado para assinatura por terceiros.	79
Figura 58 – Documento na lista de documentos já finalizado e que pode ser compartilhado por link.	80
Figura 59 – Compartilhamento por link de documento já inserido no sistema.	80
Figura 60 – Opção de desmarcar a assinatura do próprio usuário ao concluir o preparo.	81
Figura 61 – Assinatura posterior do documento pelo usuário que preparou.	82
Figura 62 – Assinatura posterior do documento pelo usuário que preparou.	82

LISTA DE ABREVIATURAS E SIGLAS

ADES	ADvanced Electronic Signature
BPMN	Business Process Model and Notation
CAFe	Comunidade Acadêmica Federada
FK	Chave estrangeira da tabela em modelagem de banco de dados
GED	Gerenciador Eletrônico de Documentos
ICP	Infraestrutura de Chaves Públicas
ICPEdu	Infraestrutura de Chaves Públicas para Ensino e Pesquisa
IdP	Provedor de Identidade
INTIC	Instituto Nacional de Tecnologias de Informação e Comunicação (Moçambique)
LabSEC	Laboratório de Segurança em Computação
OTC	One-Time Certificate
PAS	Portal de Assinatura
PDF	Portable Document Format
PdD	Preparador de Documentos
PK	Chave primária da tabela em modelagem de banco de dados
RF	Requisito Funcional
RNP	Rede Nacional de Ensino e Pesquisa
UFSC	Universidade Federal de Santa Catarina

SUMÁRIO

1	INTRODUÇÃO	15
1.1	OBJETIVOS	16
1.1.1	Objetivo Geral	16
1.1.2	Objetivos Específicos	16
1.2	METODOLOGIA	17
1.2.1	Primeira etapa: Levantamento de Requisitos	17
1.2.2	Segunda etapa: Proposta e implementação	17
1.2.3	Terceira etapa: Verificação em relação aos requisitos	18
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	ONE-TIME CERTIFICATE	19
2.2	NOTAÇÃO BPMN	19
2.3	WHATSAPP	20
2.4	FERRAMENTAS DE DESENVOLVIMENTO DE <i>SOFTWARE</i>	20
2.4.1	<i>Python e Flask</i>	20
2.4.2	<i>Javascript, Typescript e React</i>	21
2.4.3	<i>PostgreSQL</i>	21
2.4.4	<i>Vite e Node.js</i>	21
2.4.5	<i>Docker</i>	22
3	TRABALHOS RELACIONADOS	23
3.1	ASSINADOR DIGITAL AVANÇADO: PROPOSTA DE INTERFACE <i>WEB</i> DE ASSINATURA ELETRÔNICA ACESSADA POR <i>SMARTPHONE</i> . . .	23
3.2	MENOS CERTIFICAÇÃO DIGITAL E MAIS IDENTIDADE ELETRÔNICA: ICPEDU E CAFE EM UM ASSINADOR DIGITAL INCLUSIVO .	23
3.3	ASSINATURA DE DOCUMENTOS COM CERTIFICADO DE USO ÚNICO E CHAVES GERADAS NO CLIENTE	24
3.4	SISTEMA DE GESTÃO DE PESSOAS: ESPECIFICAÇÃO DE REQUISITOS FUNCIONAIS PARA OS PROCESSOS SELETIVOS SIMPLIFICADOS DA UFSC	25
4	LEVANTAMENTO DE REQUISITOS	26
4.1	FLUXOS DE USO DO ADES	26
4.1.1	Fluxo de assinatura de documento ainda não inserido no sistema	26
4.1.2	Visualização de documentos inseridos no sistema	29
4.1.3	Compartilhamento de documentos e coleta de assinaturas	29
4.1.4	Limitações no fluxo de assinaturas atual do ADES	31
4.2	REQUISITOS FUNCIONAIS DO PREPARADOR DE DOCUMENTOS . .	31

4.2.1	Objetivos de uso coletados em trabalho anterior	31
4.2.2	Requisitos coletados a partir de reuniões com gestores do projeto ADES	32
4.2.3	Tabela de requisitos funcionais	34
4.3	FLUXO DE TELAS PROPOSTO EM TRABALHO ANTERIOR	36
4.3.1	Protótipo interno desenvolvido no LabSEC	37
4.3.2	Atendimento dos requisitos no fluxo sugerido	41
4.3.3	Limitações do fluxo de telas proposto	42
5	PROPOSTA	45
5.1	ARQUITETURA DO PROJETO ADES E SERVIÇOS IMPACTADOS	45
5.1.1	Portal de Assinatura e Gerenciador Eletrônico de Documentos	46
5.2	FLUXOS DE PROCESSO DO PREPARADOR DE DOCUMENTOS	46
5.2.1	Fluxo principal: preparo, coleta e recebimento	46
5.2.2	Assinatura com um clique sem artefato visual	51
5.2.3	Compartilhamento por link sem signatários esperados	52
5.2.4	Compartilhamento por link com signatários esperados	52
5.2.5	Compartilhamento de documento já inserido no sistema	52
5.2.6	Adiar a própria assinatura	53
5.2.7	Monitoramento e estados do documento	53
6	IMPLEMENTAÇÃO	54
6.1	ADIÇÕES AO PORTAL DE ASSINATURA	54
6.2	MUDANÇAS NO GERENCIADOR ELETRÔNICO DE DOCUMENTOS	55
6.2.1	Ajustes no banco de dados	55
6.2.2	Criação de novas funções	57
7	RESULTADOS	60
7.1	DEMONSTRAÇÃO DO FLUXO PRINCIPAL	60
7.1.1	Selecionar novo documento	60
7.1.2	Adicionar signatários ao documento	63
7.1.3	Envio do documento e finalização do preparo	66
7.1.4	Assinatura do signatário a partir do e-mail	69
7.1.5	Notificação de pedido de assinatura por WhatsApp	71
7.1.6	Recebimento do documento finalizado	73
7.2	DEMONSTRAÇÃO DE COMPORTAMENTOS ALTERNATIVOS	74
7.2.1	Monitoramento do estado do documento	74
7.2.2	Documentos que não receberam todas assinaturas esperadas	75
7.2.3	Assinatura com 1 só clique sem artefato visual	78
7.2.4	Compartilhar documento por link	78
7.2.5	Compartilhar documentos já inseridos no sistema	79

7.2.6	Adiar a própria assinatura em um documento preparado em que o usuário é um signatário especificado	81
8	ATENDIMENTO AOS REQUISITOS FUNCIONAIS	83
9	CONCLUSÃO	86
9.1	TRABALHOS FUTUROS	86
	REFERÊNCIAS	87
	APÊNDICE A – CÓDIGO-FONTE	89
	APÊNDICE B – ARTIGO PARA PUBLICAÇÃO	90

1 INTRODUÇÃO

A sociedade encontra-se em uma era de crescente digitalização, onde documentos e processos que antes eram físicos agora encontram-se predominantemente no formato digital. A assinatura em documentos digitais se demonstra como uma alternativa eficaz em trazer um processo que era normalmente no papel para o ambiente virtual. No Brasil, assinaturas digitais são reconhecidas legalmente para operações jurídicas (BRASIL, 2020), criando condições para que seu uso seja gradualmente ampliado no país.

Nesse contexto está o projeto ADvanced Electronic Signature (ADES), do Laboratório de Segurança em Computação (LabSEC) da Universidade Federal de Santa Catarina (UFSC), do qual nascem sistemas digitais que permitem a assinatura, verificação e compartilhamento de documentos eletrônicos do tipo PDF, de forma simples e rápida, dependendo apenas do usuário se autenticar (LabSEC, 2023). O projeto ADES possui seu próprio sistema disponibilizado para uso, mas também serviu como base para a criação de sistemas semelhantes para outras instituições, como o Assinador Digital Avançado do Instituto Nacional de Tecnologias de Informação e Comunicação (INTIC, 2023), e o Sistema de Assinatura Eletrônica da ICPEdu (Perottoni et al., 2023). Todos esses sistemas, derivados do ADES, compartilham fluxos de interação muito semelhantes e oferecem as mesmas funcionalidades principais: assinatura, verificação e compartilhamento de documentos. O acesso a esses sistemas é realizado por meio de interfaces *web*, que podem ser acessados através de navegadores de internet.

No fluxo atual da funcionalidade de realizar assinaturas no ADES, o usuário deve se autenticar, enviar um documento, e clicar em assinar. Esse fluxo é detalhado no Capítulo 4. Para outras pessoas assinarem o documento, é necessário que elas recebam uma *Uniform Resource Locator (URL)* de compartilhamento do primeiro signatário. Entretanto, esse fluxo de uso ainda contém potencial de melhorias para atender mais casos de uso com maior eficácia. Atualmente, ele carece de mais mecanismos de controle como permitir especificar quem são os signatários. Outro ponto relevante é que o trâmite de documentos ainda demanda que a pessoa que compartilha o documento utilize também de sistemas externos para entrar em contato com o signatário e enviar o link de forma manual. Além disso, quem aguarda o recebimento das assinaturas não dispõe de uma forma objetiva e automatizada para saber se todas assinaturas que espera foram recebidas, e acabam dependentes de monitorar manualmente as assinaturas no documento ou depender de uma confirmação por mensagem do signatário. Melhorias nesses aspectos do sistema por meio de funcionalidades que apoiem nesses aspectos podem aumentar a eficiência do fluxo para os mais diversos tipos de utilizadores do sistema, como pessoas encarregadas de preparar documentos e coletar assinaturas.

Diante desse contexto e inspirado em tornar o ADES mais competitivo em relação a assinadores que oferecem algumas funcionalidades que permitem o preparo de documentos para recebimento de assinaturas como o Assin@Ufsc ou *DocuSign* (Lima, 2023), surge a demanda de funcionalidades no sistema ADES para preparar documentos para receberem assinaturas digitais e para monitorá-los, nomeada como Preparador de Documentos (PdD). Ele busca suprir

essas lacunas com novos mecanismos de controle para quem prepara documentos e coleta assinaturas e também automatizações no processo de comunicação entre as partes. O projeto, que continuamente cresce, beneficia-se de que novas funcionalidades estejam alinhadas aos requisitos acordados e verificáveis antes de disseminação. Isso mostra-se relevante uma vez que o LabSEC mantém parcerias e colaborações com as instituições que utilizam seus sistema derivados, podendo eventualmente cedê-las código-fonte com as funcionalidades implementadas ou mesmo servir como inspiração para elas realizarem melhorias semelhantes, e qualquer alteração em funcionalidades pode gerar consequências na experiência de uso do sistema que afetarão positivamente ou negativamente o dia a dia de milhões de usuários, como os do sistema do INTIC, que é disponibilizado para uso à toda população de Moçambique (INTIC, 2023), estimada em 33 milhões em 2022 (World Bank Group, 2025), ou então do sistema fornecido à ICPEdu, que o disponibiliza aos mais de 1 milhão de membros da Comunidade Acadêmica Federada (CAFe) (Perottoni et al., 2023).

Assim, este trabalho busca materializar o fluxo de preparação de documentos do PdD, com base no trabalho de Lima (2023) e requisitos funcionais coletados, a fim de fornecer novos mecanismos para o preparo e gerenciamento de documentos eletrônicos, e demonstrar que a implementação atende a esse conjunto de requisitos por meio de evidências visuais da aplicação.

1.1 OBJETIVOS

Este trabalho define o que se pretende alcançar com a implementação e a verificação do PdD no sistema ADES em um objetivo geral e em objetivos específicos.

1.1.1 Objetivo Geral

Implementar o PdD no sistema de assinatura digital de documentos PDF do projeto ADES e verificar o atendimento aos requisitos funcionais definidos, com evidências na aplicação.

1.1.2 Objetivos Específicos

- Levantar requisitos funcionais do PdD com base na análise das demandas atuais do projeto ADES e nos objetivos de uso coletados no trabalho de Lima (2023);
- Implementar o PdD no sistema do projeto ADES com base nos requisitos funcionais definidos e fluxos de telas criados no trabalho de Lima (2023);
- Verificar o atendimento aos requisitos funcionais por meio do cruzamento entre cada requisito e o comportamento implementado, documentando evidências nas telas do sistema.

1.2 METODOLOGIA

A condução do trabalho segue uma abordagem em etapas alinhada aos objetivos propostos: levantam-se os requisitos, define-se a proposta de inserção do PdD na arquitetura do ADES, realiza-se a implementação no ecossistema do projeto e, por fim, verificam-se os resultados e o atendimento aos requisitos funcionais por meio de evidências na aplicação.

1.2.1 Primeira etapa: Levantamento de Requisitos

Os requisitos funcionais para implementação do PdD serão definidos com base em demandas recebidas através de reuniões com gestores do projeto ADES, considerando também os objetivos de uso no trabalho de Lima (2023), além de novos requisitos levantados que atendam a casos de uso que possam não terem sido cobertos anteriormente.

Em seguida, será realizada uma análise do fluxo de telas proposto no trabalho de Lima (2023) para verificar como ele atende aos requisitos funcionais definidos, identificando quais requisitos são adequadamente cobertos e quais necessitam de ajustes ou complementações no design proposto.

1.2.2 Segunda etapa: Proposta e implementação

Com os requisitos definidos, elabora-se a proposta de inserção do PdD na arquitetura do ADES e a modelagem dos fluxos de processo (Capítulo 5). Em seguida, realiza-se a implementação no PAS e no GED (Capítulo 6). Para implementação do sistema, as ferramentas escolhidas levam em conta as que já são adotadas no projeto para lógica do lado do navegador e do servidor, a fim de manter padronização. A lógica do lado do navegador será feita com a linguagem de programação *Javascript* (JavaScript Team at Mozilla, 2025) acompanhada das bibliotecas *React* (React Team at Meta, 2025), para obter uma página com elementos responsivos que mudam dinamicamente, e *Typescript* (TypeScript Team, 2025), para ter um código com tipagem clara a fim de contribuir com legibilidade e manutenibilidade. A página utiliza a linguagem de marcação *HTML* (Mozilla Developer Network, 2025b) e é estilizada com *CSS* (Mozilla Developer Network, 2025a). Para o processamento dos arquivos do fluxo de preparo de documentos, a aplicação em execução no navegador deverá realizar requisições *HTTP* ao servidor contendo os arquivos ou informações para processá-los. Para compilar a aplicação *frontend* e gerar os arquivos que serão usados em ambiente de produção, será utilizada a ferramenta *Vite* (Vite Team, 2025), enquanto *Node.js* (Node.js Foundation, 2025) será utilizado para servir a aplicação compilada em ambiente de produção. Já funcionalidades no servidor serão implementadas em *Python* (Python Software Foundation, 2025), com o módulo *Flask* (Pallets Team, 2023) para permitir que a aplicação receba requisições *HTTP* em suas rotas definidas e retorne respostas processadas ao cliente (navegador do usuário). Para guardar de forma eficiente dados relacionados aos documentos inseridos no sistema, o programa em *Python* utilizará

o módulo *Psycopg* (The Psycopg Team, 2025) para manipular um banco de dados *PostgreSQL* (PostgreSQL Global Development Group, 2025), o qual armazenará os documentos e outras informações. Os microsserviços do ecossistema ADES são empacotados e executados em contêineres *Docker* (Docker Inc., 2025), tecnologia já empregada no projeto para padronizar ambientes de desenvolvimento e produção. A estrutura implementada no PAS e no GED é detalhada no Capítulo 6.

1.2.3 Terceira etapa: Verificação em relação aos requisitos

Com a implementação concluída, o próximo passo será a verificação de que o sistema atende aos requisitos funcionais elencados. Cada requisito será relacionado ao comportamento observado na aplicação e a telas ou fluxos correspondentes, conforme será registrado no Capítulo 7 e sintetizado em uma tabela no Capítulo 8. Essa etapa caracterizará uma verificação funcional e de rastreabilidade em relação ao que tiver sido especificado.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 ONE-TIME CERTIFICATE

As assinaturas digitais realizadas no ADES apoiam-se na tecnologia de One-Time Certificate (OTC). Esse tipo de certificado digital (Mayr et al., 2024; Schardong; Mayr; Custódio, 2022) surgiu para resolver desafios de gerenciamento de chaves privadas nas infraestruturas de chave pública (ICPs), especificamente para assinatura digital de documento por pessoas, sendo aplicada em sistemas de assinaturas digitais como o ADES. Fazer com que o usuário final precise gerenciar sua chave privada de forma segura abre portas para problemas de confiança e usabilidade e que levantam preocupações relacionadas à privacidade.

Para resolver esses desafios foi proposto o OTC, que é um certificado digital que dispensa a necessidade de armazenamento de chaves privadas, de revogação de certificados e de uso do carimbo do tempo. É concebido que um OTC seja criado para cada assinatura digital e usado em um único documento, o que é garantido por mecanismos como incluir nele o resumo criptográfico do documento. Assim, no processo de assinaturas digitais, coloca-se a responsabilidade de autenticar o assinante no Provedor de Identidade (IdP), serviço que verifica a identidade do usuário (por exemplo, por meio de credenciais institucionais) e fornece ao sistema os atributos necessários para vinculá-lo à assinatura.

Percebe-se que grandes comunidades podem ter custos reduzidos para fornecer assinaturas digitais usando esse sistema e com melhor segurança do que em sistemas tradicionais, dado não precisarem armazenar as chaves (Mayr et al., 2024). OTCs são utilizados nos sistemas de assinaturas digitais originados do projeto ADES (Perotoni et al., 2023). Em comparação a assinadores baseados em ICP tradicional, a simplificação para o usuário final não se manifesta necessariamente em menos etapas na interface, mas na redução de barreiras técnicas preparatórias: o usuário não precisa obter, instalar ou gerenciar certificado digital nem chave privada (por exemplo, em *token* ou *smart card*), nem lidar com revogação de certificados; basta autenticar-se no IdP, enviar o documento e solicitar a assinatura. Assim, decisões de usabilidade, design e implementação para o fluxo de preparo e assinatura de documentos podem ser tomadas tendo em mente que a comprovação de identidade do signatário concentra-se na autenticação institucional, e não no manuseio de credenciais criptográficas pelo usuário.

2.2 NOTAÇÃO BPMN

O *Business Process Model and Notation* (BPMN) foi desenvolvido com o objetivo principal de prover uma notação que seja prontamente compreensível por todos os usuários de negócio, preenchendo a lacuna de comunicação que frequentemente existe entre o desenho de um processo e sua implementação técnica (White, 2004). O BPMN utiliza uma técnica de modelagem baseada em fluxogramas, porém adaptada para criar modelos gráficos de operações de processos de negócio, permitindo que analistas, desenvolvedores e gerentes compartilhem

uma mesma linguagem visual para monitorar e gerenciar fluxos de trabalho.

A estrutura visual da BPMN fundamenta-se em elementos que organizam responsabilidades e controlam a lógica do fluxo. A organização dos processos é feita através de *Pools* (Piscinas), que representam participantes ou entidades distintas, e *Lanes* (Raias), que subdividem a *Pool* para indicar papéis ou departamentos específicos. As unidades de trabalho são representadas por atividades (*Tasks*), que no contexto de sistemas de informação, costumam ser classificadas como *User Tasks* (atividades que dependem de interação humana) ou *Service Tasks* (atividades automatizadas executadas por *software*). Já o controle de decisões é efetuado pelos *Gateways*, como o exclusivo (*XOR*), que permite que apenas um caminho seja seguido conforme uma condição, enquanto o inclusivo (*OR*) possibilita a ativação de um ou mais caminhos simultâneos (White, 2004). Neste trabalho, o BPMN é utilizado para modelar os fluxos de processo do PdD no Capítulo de Proposta (Seção 5.2), permitindo uma visualização clara e compreensível do preparo e da coleta de assinaturas.

2.3 WHATSAPP

O WhatsApp é um aplicativo de mensagens instantâneas disponível para celulares, usado por bilhões de pessoas em diversos países para trocar mensagens, mídias e realizar ligações de voz de forma simples e confiável (WhatsApp LLC, 2026). Inicialmente concebido como alternativa ao SMS, o serviço suporta o envio e recebimento de texto, fotos, vídeos, documentos e localização, com criptografia de ponta a ponta nas conversas (WhatsApp LLC, 2026). Além do uso direto pelo aplicativo, a plataforma disponibiliza uma API que permite que outras aplicações enviem mensagens aos usuários, o que viabiliza notificações automatizadas em fluxos como o de pedido de assinatura tratado neste trabalho.

2.4 FERRAMENTAS DE DESENVOLVIMENTO DE SOFTWARE

As tecnologias empregadas na implementação são as já adotadas no projeto ADES, escolhidas para manter padronização entre o portal e o servidor.

2.4.1 Python e Flask

Python é uma linguagem de programação interpretada, orientada a objetos, de alto nível, e com semântica dinâmica. Destaca-se para o uso em desenvolvimento ágil de aplicações pois permite criar programas de fácil manutenção de forma eficiente e com alta legibilidade, dado características como tipagem dinâmica e vinculação flexível de variáveis, sintaxe simples e intuitiva, e modularidade e reaproveitamento de código por meio de suporte a módulos e pacotes (Python Software Foundation, 2025).

O *Flask* é uma *framework* leve para aplicações *web* em *Python* que oferece ferramentas importantes para construir esse tipo de aplicação, como roteamento de *URLs* e manipulação de

requisições e respostas *HTTP* (Pallets Team, 2023). Pode ser utilizado diretamente em um programa feito em *Python* por meio da importação desse módulo. Seu caso de uso principal é expor rotas da aplicação que, ao receberem requisições de clientes, executam operações e retornam uma resposta.

2.4.2 *JavaScript, Typescript e React*

JavaScript é uma linguagem de programação criada originalmente para executar no navegador, trabalhando em conjunto com *HTML* (estrutura) e *CSS* (estilo) para fornecer páginas *web* funcionais (JavaScript Team at Mozilla, 2025).

TypeScript (TypeScript Team, 2025) é uma linguagem estendida de *JavaScript*, que adiciona tipagem estática e recursos avançados de desenvolvimento. Ele permite escrever programas mais robustos, com mais regras e verificações estáticas do código.

React é uma biblioteca a base de *JavaScript* de código aberto, utilizada junto ao *JavaScript* para construir interfaces de usuário dinâmicas e eficientes, especialmente em aplicações que alteram dinamicamente elementos da tela sem atualizar a página no navegador (React Team at Meta, 2025).

2.4.3 *PostgreSQL*

O *PostgreSQL* é um sistema gerenciador de banco de dados objeto-relacional que usa e estende a linguagem de gerenciamento de dados relacionais *SQL* (PostgreSQL Global Development Group, 2025). Bancos de dados relacionais surgem como uma forma segura de guardar grandes quantidades de dados em tabelas, com linhas e colunas e métodos eficientes de consulta (Codd, 1970). Para manipulação do banco, o módulo *Psycopg* (The Psycopg Team, 2025) permite a softwares baseados na linguagem *Python* ferramentas para a manipulação de banco de dados objeto-relacional baseados em *PostgreSQL*.

2.4.4 *Vite e Node.js*

Vite é uma ferramenta moderna de compilação para aplicações *frontend* que oferece um ambiente de desenvolvimento rápido e eficiente, além de otimizar o processo de geração dos artefatos para produção. Ele processa e otimiza o código *JavaScript*, *TypeScript* e outros recursos da aplicação, gerando arquivos estáticos otimizados que podem ser servidos em ambiente de produção (Vite Team, 2025).

Node.js é um ambiente de execução *JavaScript* que permite executar código *JavaScript* no servidor. No contexto de aplicações *web*, *Node.js* é frequentemente utilizado para servir aplicações *frontend* já compiladas, fornecendo um servidor *HTTP* que distribui os arquivos estáticos gerados pelo processo de compilação para os clientes que acessam a aplicação (Node.js Foundation, 2025).

2.4.5 *Docker*

Docker é uma plataforma de containerização que permite empacotar aplicações e suas dependências em ambientes isolados chamados *containers* (Docker Inc., 2025). A containerização é uma tecnologia que encapsula uma aplicação junto com todas as suas dependências (bibliotecas, ferramentas, configurações) em um ambiente isolado e portátil. Esses *containers* são unidades leves e independentes que podem ser executados em qualquer sistema que tenha o *Docker* instalado, garantindo que a aplicação funcione de forma consistente independentemente do ambiente onde é executada, seja em desenvolvimento, teste ou produção.

A criação desses *containers* é definida através de arquivos chamados *Dockerfile*, que contêm instruções passo a passo para construir o que é conhecido como uma imagem. Uma imagem *Docker* é um modelo ou *template* que descreve como o *container* deve ser configurado, incluindo qual sistema operacional base utilizar, quais dependências instalar, quais comandos executar durante a construção da imagem e como a aplicação deve ser iniciada. Quando uma imagem é executada, ela se torna um *container* em execução.

3 TRABALHOS RELACIONADOS

3.1 ASSINADOR DIGITAL AVANÇADO: PROPOSTA DE INTERFACE *WEB* DE ASSINATURA ELETRÔNICA ACESSADA POR *SMARTPHONE*

No trabalho de Lima (2023), em parceria com o LabSEC, começou-se a trabalhar no PdD do projeto ADES. Após reuniões com membros do projeto, definiram-se requisitos funcionais que, juntos a pesquisas com o público-alvo e diversas iterações de prototipagem e testes com usuários em protótipos, resultaram em um protótipo de fluxo de telas e interações do usuário com o sistema, que foi entregue ao laboratório, acompanhados de proposta de identidade visual para estilização dos elementos.

Entretanto, o fluxo proposto deixa pendências que abrem portas para o presente trabalho. Isso porque não foi de fato implementada e integrada no sistema ADES, nem foi proposta como seria para telas de computador (*desktop*). Além disso, desde a realização do trabalho de Lima (2023), surgiram novos requisitos funcionais mais específicos para o PdD ser implantado no ADES e que exigem adaptações no fluxo proposto, que envolvem melhorias que visam cobrir mais casos de uso e que serão tratados neste trabalho.

3.2 MENOS CERTIFICAÇÃO DIGITAL E MAIS IDENTIDADE ELETRÔNICA: ICPEDU E CAFE EM UM ASSINADOR DIGITAL INCLUSIVO

O artigo de Perottoni et al. (2023) apresenta a arquitetura de um sistema de assinatura digital voltado aos membros da CAFe, com o objetivo de oferecer um processo desburocratizado ao usuário final, sem interação direta com a infraestrutura de certificação digital da ICPEdu. O texto descreve uma arquitetura de microsserviços (Perottoni et al., 2023) cujos componentes incluem o PAS, verificador de assinaturas, assinador e gerenciador de arquivos, entre outros. Para o usuário, a experiência de assinatura consiste em autenticar-se, inserir o documento PDF a ser assinado no sistema e clicar para assinar, sem se preocupar com gestão de certificados e de chaves.

O assinador digital da RNP é desenvolvido em parceria com o LabSEC como sistema derivado do ADES, compartilhando tecnologias e arquitetura semelhantes, incluindo o uso de OTC e autenticação via IdP. A relação desse artigo com o presente trabalho não se estabelece por convergência de requisitos funcionais de preparo documental, uma vez que o texto não trata do fluxo de preparo, da especificação de múltiplos signatários, de prazos nem de notificações. Trata-se, antes, de referência da infraestrutura arquitetural na qual o PdD foi integrado: os microsserviços descritos, em especial o PAS e o GED, constituem a base do ecossistema estendida pelas funcionalidades de preparo e coleta de assinaturas desenvolvidas neste trabalho.

3.3 ASSINATURA DE DOCUMENTOS COM CERTIFICADO DE USO ÚNICO E CHAVES GERADAS NO CLIENTE

O trabalho de Cabral (2023) apresenta o desenvolvimento de uma aplicação *web* voltada para a realização de assinaturas digitais em documentos no formato PDF utilizando OTCs. A motivação apresentada está relacionada à simplificação do acesso a assinaturas digitais, reduzindo barreiras associadas ao gerenciamento de chaves privadas convencionais (como o uso de *smart cards* ou *tokens*) e a problemas de revogação de certificados em infraestruturas tradicionais de ICP.

Uma contribuição central desse trabalho é a preservação da privacidade do usuário por meio da execução das operações criptográficas no navegador. No fluxo original do projeto ADES, o processamento e a assinatura ocorriam no servidor, exigindo o envio do arquivo para o *backend*. Na solução proposta por Cabral (2023), a geração do par de chaves assimétricas e a manipulação da estrutura do PDF ocorrem diretamente no navegador do cliente. Para isso, a aplicação emprega a *Web Crypto API* nativa do navegador e bibliotecas como *pkij*s e *node-forge* para realizar atualizações incrementais no PDF e inserir o *placeholder* da assinatura. O servidor atua apenas como intermediário que encaminha a requisição de certificado (*Certificate Signing Request*, CSR) contendo a chave pública e o resumo criptográfico (*hash*) do documento para o microserviço de ICP, que emite o certificado OTC correspondente. Desse modo, o arquivo binário do documento não é enviado ao servidor durante o processo criptográfico, o que busca resguardar o sigilo dos dados.

A relação entre o trabalho de Cabral (2023) e este trabalho estabelece-se a partir do ecossistema compartilhado de soluções do LabSEC voltadas ao projeto ADES. Ambos compartilham a premissa de simplificar a experiência do usuário final utilizando OTCs e arquiteturas de microserviços baseadas em *containers Docker*. No entanto, enquanto Cabral (2023) concentra-se nos aspectos de segurança, privacidade local e viabilidade técnica da execução criptográfica no lado do cliente para um fluxo de assinatura individual, o presente trabalho expande o escopo para a coordenação de fluxos colaborativos com múltiplos signatários.

Em suas conclusões, Cabral (2023) aponta como trabalho futuro a elaboração de um sistema de gerenciamento de assinaturas, com possibilidade de convidar pessoas a assinarem um mesmo documento e de acompanhar quais usuários requisitados já assinaram. Essa recomendação alinha-se às lacunas de controle e de coordenação do trâmite documental já identificadas na Introdução para o fluxo original do ADES, como especificar signatários, monitorar o recebimento de assinaturas e apoiar a comunicação entre as partes, e constitui uma das motivações do PdD tratado neste trabalho, cujos requisitos funcionais são sistematizados no Capítulo 4. Portanto, a presente pesquisa complementa a infraestrutura de assinatura no cliente proposta por Cabral (2023) com a camada de gerenciamento de fluxos documentais necessária para a evolução do ecossistema ADES.

3.4 SISTEMA DE GESTÃO DE PESSOAS: ESPECIFICAÇÃO DE REQUISITOS FUNCIONAIS PARA OS PROCESSOS SELETIVOS SIMPLIFICADOS DA UFSC

O trabalho de Schmitz (2018) trata da especificação de requisitos funcionais para um sistema de gestão de Processos Seletivos Simplificados de professores substitutos na UFSC, sob a perspectiva da administração universitária e da gestão de pessoas. A pesquisa mapeia fluxos de trabalho institucionais com participação de diversos atores e propõe a substituição de controles manuais descentralizados por um sistema de informação integrado, formalizando as funcionalidades esperadas em documento de requisitos.

Há paralelos com esta pesquisa no tratamento de fluxos complexos com múltiplos participantes e na consolidação de requisitos funcionais a partir de regras de negócio institucionais. Em Schmitz (2018), isso se manifesta na modelagem de etapas administrativas e na identificação de responsabilidades de cada ator envolvido no processo seletivo. No PdD, o mesmo tipo de preocupação aparece na definição de papéis como preparador e signatário, no acompanhamento do *status* das assinaturas e nas regras que regem prazos, notificações e conclusão do documento.

Entretanto, os domínios são distintos: Schmitz (2018) focaliza processos seletivos de pessoal, enquanto o presente trabalho trata da preparação e da coleta de assinaturas em documentos no ADES. Outra diferença relevante é o alcance do ciclo de desenvolvimento: Schmitz (2018) limita-se à especificação de requisitos, sem implementação ou verificação em sistema em operação. Neste trabalho, o levantamento de requisitos do PdD é seguido de proposta arquitetural, implementação no PAS e no GED, e verificação do atendimento dos requisitos por meio de evidências na aplicação.

4 LEVANTAMENTO DE REQUISITOS

4.1 FLUXOS DE USO DO ADES

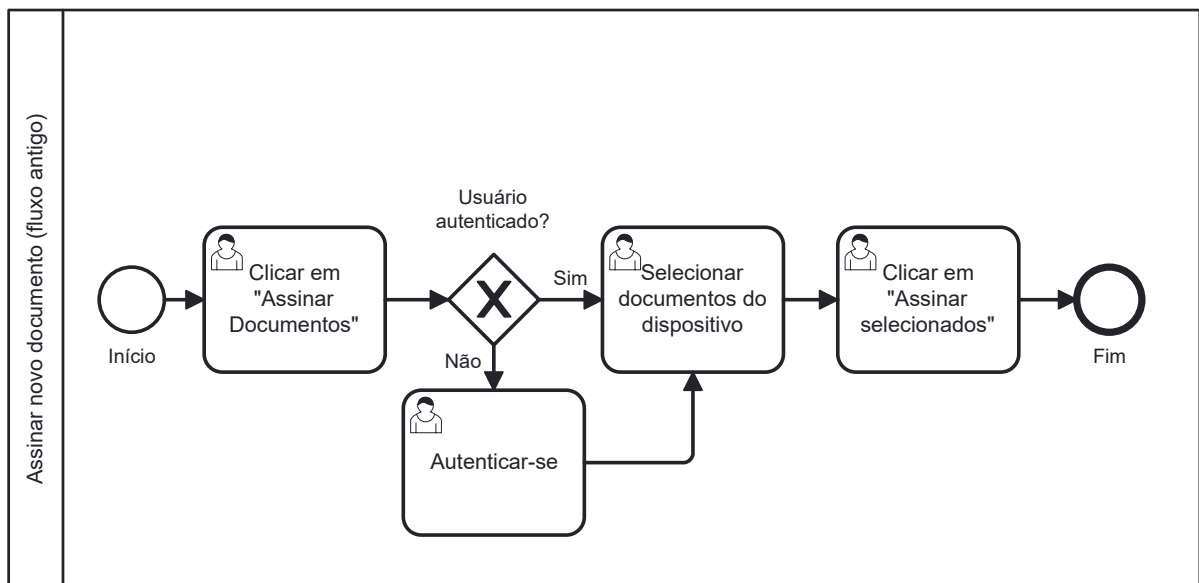
O sistema ADES, atualmente em operação¹, conta com diversas funcionalidades já implementadas e fluxos claros para realização de assinatura em documentos digitais.

4.1.1 Fluxo de assinatura de documento ainda não inserido no sistema

O fluxo principal para assinatura, a partir da página principal, consiste em se autenticar, inserir um documento no sistema e enviá-lo para assinatura. A Figura 1 mostra o processo detalhado, na forma de um fluxo BPMN. Consiste em:

- Clicar em "Assinar documentos" (Figura 2);
- Caso não autenticado, é redirecionado à tela de autenticação (Figura 3), no qual precisa se autenticar perante um IdP;
- Ao ser redirecionado para a página de inserção de documentos, clicar em "procurar" e selecionar o documento PDF a ser assinado (Figura 4);
- Clicar no botão de assinatura e aguardar a realização da assinatura (Figura 5).

Figura 1 – Processo de assinatura novo documento no fluxo antigo.



Fonte: elaborado pelo autor.

¹ <https://ades.labsec.ufsc.br>

Figura 2 – Tela da página principal.



Fonte: elaborado pelo autor.

Figura 3 – Tela de autenticação.

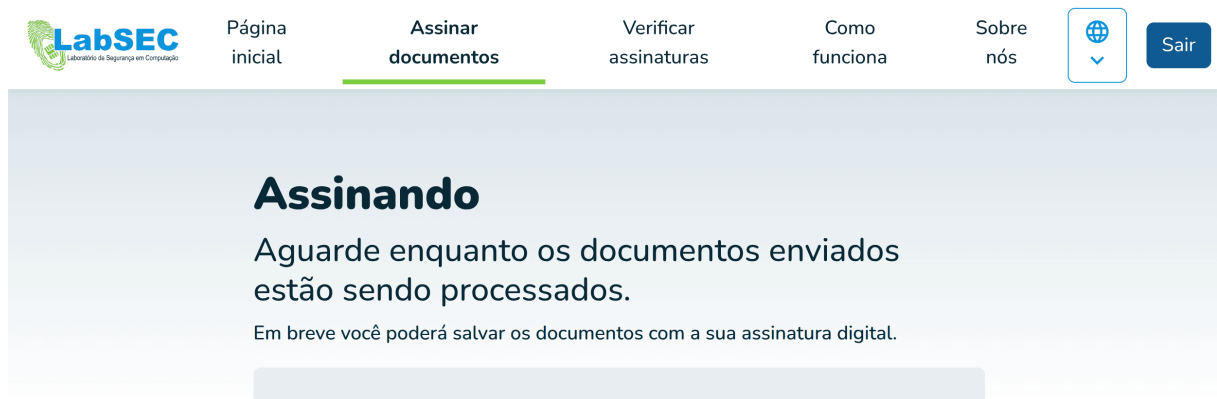
Fonte: elaborado pelo autor.

Figura 4 – Tela com documento selecionado para assinatura.



Fonte: elaborado pelo autor.

Figura 5 – Tela de progressão do processo de assinatura.

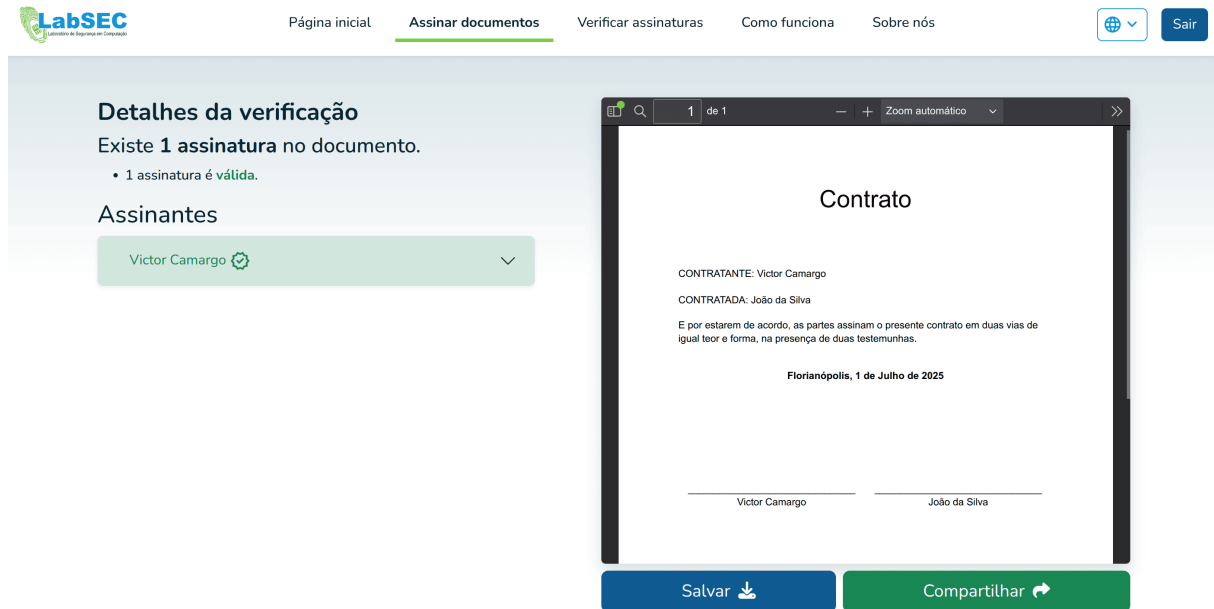


Fonte: elaborado pelo autor.

4.1.2 Visualização de documentos inseridos no sistema

O documento inserido no sistema por meio do fluxo de assinatura é exibido ao final do processo, com a assinatura listada à esquerda e o documento à direita (Figura 6)

Figura 6 – Tela de visualização do resultado da assinatura.

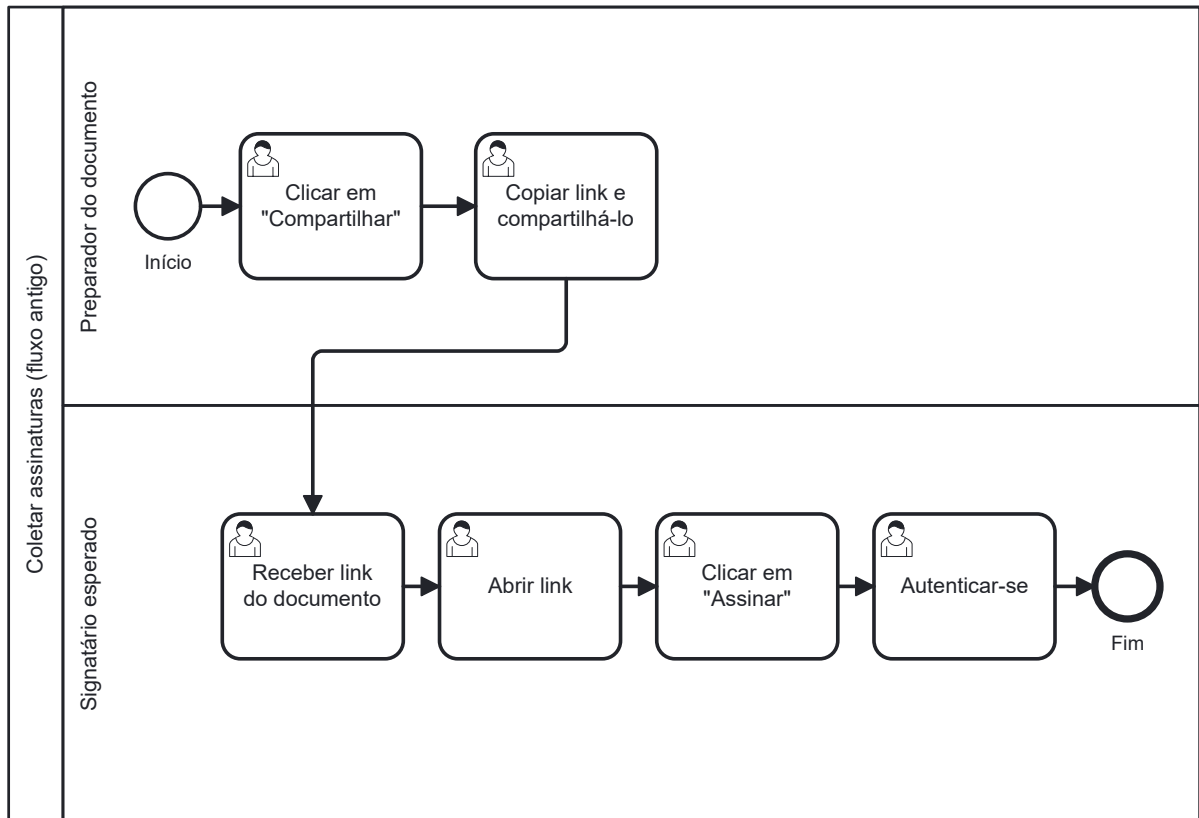


Fonte: elaborado pelo autor.

4.1.3 Compartilhamento de documentos e coleta de assinaturas

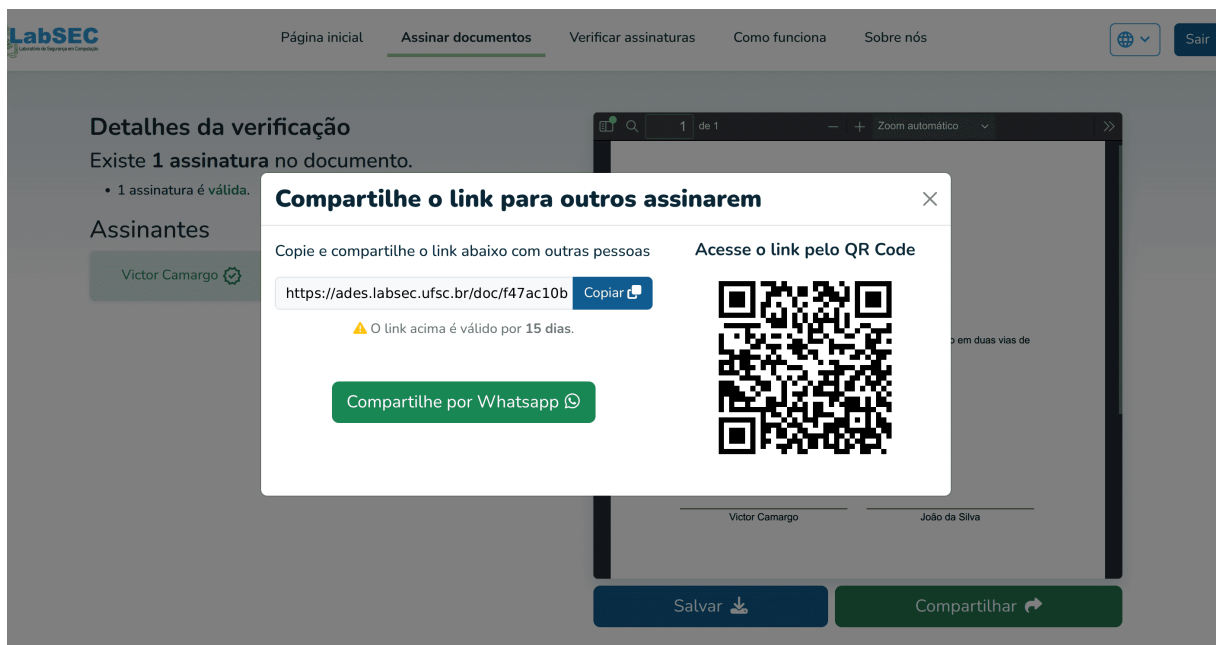
O documento pode então ser compartilhado ao clicar no botão de "Compartilhar". Isso resultará na geração de um novo link (Figura 8), que deve ser repassado a usuários que queiram visualizá-lo e baixá-lo, sem necessidade de autenticação, ou assiná-lo, o que requer autenticação. A Figura 7 mostra o fluxo que um usuário pode seguir para realizar o compartilhamento de documentos e coleta de assinaturas de outros signatários dentro do sistema.

Figura 7 – Mapa de navegação do fluxo de compartilhamento de documentos e coleta de assinaturas.



Fonte: elaborado pelo autor.

Figura 8 – Tela ao clicar em compartilhar com documento selecionado e autenticado.



Fonte: elaborado pelo autor.

4.1.4 Limitações no fluxo de assinaturas atual do ADES

Por mais que a abordagem atual forneça um fluxo simples e eficiente para a realização de assinaturas digitais e compartilhamento de documentos, com telas com mensagens claras sobre as operações que podem ser realizadas que não requerem que o usuário tenha conhecimento profundo sobre assinaturas digitais, a análise do fluxo em operação identificou as limitações a seguir, que restringem casos de uso relevantes e motivam a evolução do ADES em direção a um produto mais completo e competitivo no mercado de sistemas de assinatura digital.

A primeira limitação identificada é a ausência de um fluxo claro para inserir documentos no sistema sem que quem os insere precise também assiná-los. Assim, pessoas responsáveis por coletar assinaturas acabam sendo obrigadas a ou assinar o documento quando não precisam para inseri-lo no sistema ou pedir isso à pessoa na qual se espera a assinatura, o que exige o uso de ferramentas externas para comunicação e coordenação entre as partes envolvidas, tornando o processo mais lento.

A segunda limitação diz respeito à coleta de assinaturas de múltiplos signatários específicos: o sistema não permite especificar de quem se espera assinaturas ao compartilhar um documento, o que torna o processo de coleta menos eficiente e impede o monitoramento claro da progressão das assinaturas esperadas. O responsável por aguardar pelas assinaturas precisa verificar manualmente se todas as assinaturas desejadas foram coletadas na página do documento no serviço ou depender de confirmação por mensagem do signatário para considerá-lo concluído. Como consequência, não há base no sistema para identificar e contatar os signatários esperados, de modo que meios externos precisam ser utilizados para notificá-los.

A terceira limitação ocorre após o compartilhamento do documento: qualquer usuário com o link pode visualizá-lo, baixá-lo e, caso autenticado, assiná-lo, o que gera problemas de privacidade e de controle sobre quem participa do fluxo.

4.2 REQUISITOS FUNCIONAIS DO PREPARADOR DE DOCUMENTOS

Assim, o PdD surge propondo um novo fluxo de assinaturas com diversas funcionalidades e mecanismos de controle para atender com eficiência mais casos de uso para o sistema. Esta seção apresenta o levantamento que fundamenta sua implementação e consolida o conjunto de requisitos funcionais na Tabela 1.

4.2.1 Objetivos de uso coletados em trabalho anterior

Os objetivos de uso coletados por Lima (2023) no desenvolvimento das interfaces visuais de seu trabalho oferecem uma visão sucinta das funcionalidades desejadas:

- Cadastrar signatários ao documento sendo preparado;

- Permitir que quem prepara o documento também realize sua própria assinatura de forma facilitada durante o preparo;
- Poder preparar documentos sem implicar na realização da própria assinatura;
- Acompanhar *status* das assinaturas do documento preparado;

4.2.2 Requisitos coletados a partir de reuniões com gestores do projeto ADES

Para esclarecer os objetivos de uso e transformá-los em requisitos funcionais mais precisos e menos gerais, em reuniões internas do projeto ADES no LabSEC discutiu-se, com gestores do projeto, o que é esperado das funcionalidades do PdD.²

Uma funcionalidade de grande importância é notificar os envolvidos sobre etapas relevantes do processo, como o signatário quando um documento aguarda sua assinatura e o usuário que prepara o documento quando todas as assinaturas tiverem sido recebidas.

Deseja-se, ainda, que seja possível especificar, no fluxo com signatários previamente definidos, a posição do artefato visual associado à assinatura de cada signatário no documento PDF.

Para permitir que o usuário que prepara o documento também assine sem necessariamente precisar posicionar um artefato visual, o sistema deve oferecer uma forma de assinar com um só clique após inserir o documento. Ainda assim, o usuário pode optar por adicionar-se como signatário esperado, caso deseje ter um artefato visual associado à sua assinatura. Nesse cenário, deve ser possível indicar que ele não assine o documento ao concluir o preparo, permitindo que realize sua assinatura posteriormente. Isso cobre, por exemplo, situações em que o usuário deseja primeiro coletar as assinaturas de outros signatários e, apenas ao final, realizar a sua própria assinatura.

A especificação de uma data limite para recebimento de assinaturas também é necessária, uma vez que processos de trâmite de documentos podem ter prazos para seu recebimento assinado.

Além disso, é necessário permitir ao usuário que prepara o documento especificar uma mensagem a ser enviada juntamente à notificação ao signatário, bem como informações que deixem claro o comprometimento esperado da assinatura. O objetivo é acrescentar contexto e melhorar a compreensão do signatário sobre o que estará assinando. Pensando em praticidade ao usuário, deseja-se também que, após autenticação no sistema, o signatário consiga assinar o documento com um só clique.

O PdD prevê dois modos principais de coleta de assinaturas, que podem ser escolhidos conforme o caso de uso.

No **fluxo restrito**, o preparador cadastra signatários esperados, posiciona os artefatos visuais no PDF e envia o documento sem torná-lo acessível por link público. Nesse modo, ape-

² Informações da seção baseadas em discussões internas do projeto ADES no LabSEC com gestores do projeto, em especial o Prof. Dr. Ricardo Felipe Custódio, durante o 1º semestre de 2026, por meio de conversas em videoconferências.

nas quem preparou o lote e os signatários esperados, após autenticação no IdP e correspondência do e-mail cadastrado, podem visualizar e assinar o documento. O acesso ocorre preferencialmente por meio das notificações enviadas pelo sistema (e-mail ou WhatsApp), embora também seja disponibilizado um link individual para cada convite.

No **fluxo por compartilhamento de link**, o preparador obtém um link único para encaminhar por meios externos. Qualquer pessoa que possua esse link e se autentique no sistema pode visualizar o documento e assiná-lo, sem lista fechada de convidados. Esse modo atende a trâmites abertos, nos quais não se exige cadastro prévio de signatários; nesse caso, quem acessa pelo link não pode definir a posição do artefato visual no PDF, pois posicionar assinaturas posteriormente poderia cobrir dados sensíveis do documento.

Entre esses extremos, há um **cenário combinado** frequente na prática: o preparador especifica signatários esperados (com nome, meio de contato, comprometimento e posição do artefato visual) e, ao concluir o preparo ou posteriormente, também compartilha o documento por link. Nessa situação, o link amplia o acesso: qualquer usuário autenticado que o possua pode assinar, não apenas os cadastrados. Paralelamente, o sistema mantém o registro dos signatários esperados: ao autenticar-se, verifica-se se o e-mail informado pelo IdP corresponde ao de algum signatário nomeado no preparo. Quando há correspondência, a assinatura é vinculada a esse registro e o artefato visual é inserido nas coordenadas predefinidas; quando não há correspondência, isto é, quem acessou pelo link não constava na lista, a assinatura é registrada como adicional, sem reutilizar posições reservadas. Assim, o compartilhamento por link não anula o controle sobre os signatários esperados: o preparador continua acompanhando quem já assinou entre os convidados, recebendo notificações automáticas e visualizando estados intermediários do lote, enquanto o link oferece um canal complementar de acesso, útil quando a notificação não chega ao destinatário ou quando convém divulgar o documento a um grupo mais amplo.

Quando o preparo não inclui signatários esperados e o documento segue apenas por link, a conclusão do lote ocorre ao chegar a data limite, pois não há lista fixa de assinaturas para caracterizar o encerramento antecipado. Exige-se, ainda, a possibilidade de tornar compartilhável por link um documento já inserido no sistema que não tenha sido configurado assim no momento do preparo.

Além dos objetivos de uso coletados até então, este trabalho identifica outros casos de uso que podem impactar a experiência dos usuários e precisam ser pensados. A primeira questão que surge é o caso do que deve acontecer quando um documento preparado que tem signatários especificados esperados não recebe todas as assinaturas esperadas até o prazo limite estipulado para recebê-las. Nesse caso, propõe-se que o sistema notifique o usuário que preparou o documento sobre a situação, mas ainda o permita de realizar o download do documento a partir do sistema para continuação do trâmite do documento. Para evitar que o usuário por acidente baixe o documento achando que foi concluído, um aviso ao clicar o botão de download deve ser exibido deixando clara a situação. Nesse cenário, a exibição desse documento no sistema precisa ter textos claros informando a situação de documento incompleto.

A próxima questão é que pode acontecer de um usuário preparar um documento com

diversos signatários esperados e em um momento do processo perceber que um signatário não é mais necessário no processo ou ele não poderá assinar o documento. Nesse caso, propõe-se que o sistema permita a realização do download do documento com as assinaturas recebidas até então para que o usuário que preparou possa continuar sua tramitação com os signatários restantes. Assim como na questão anterior, um aviso adequado ao clicar o botão de download deve complementar a situação ao usuário para evitar que ele baixe um documento incompleto por acidente e ache que está concluído.

Quanto ao prazo limite para recebimento de assinaturas, para evitar ambiguidade operacional, a data limite é tratada como válida até o fim do respectivo dia.

No fluxo com signatários especificados, o sistema notifica-os automaticamente pelo meio de contato informado no cadastro (e-mail ou número de telefone para WhatsApp), disponibilizando a cada convidado um link de acesso individual. Esse link contém um token único, gerado para aquele signatário e aquele documento, de modo que, ao abri-lo no navegador, o sistema possa reconhecer quem deve assinar sem exigir uma nova autenticação manual no IdP. Quando o lote também é compartilhado por link público, esse canal complementa as notificações: os signatários esperados podem assinar tanto pelo convite quanto pelo link genérico, e o sistema reconhece ambos os casos para posicionar corretamente a assinatura quando o e-mail autenticado coincide com o cadastro.

Para o caso em que o preparador é o único signatário esperado e também assina o documento, o próximo passo que o usuário tende a desejar é baixá-lo, uma vez que não há mais trâmites que podem ser feitos diretamente pelo sistema em relação a ele. Assim, buscando minimizar a quantidade de passos que o usuário precisa realizar e aumentar a clareza do sistema, propõe-se que o botão de download já seja exibido ao final do fluxo de preparo, após a finalização do processo, junto a mensagem de sucesso no processo.

Sobre o ciclo de vida do documento, é importante que o sistema trate e apresente ao usuário estados intermediários do processo de coleta de assinaturas: aguardando assinaturas, completo com todas assinaturas esperadas e incompleto (isto é, quando o prazo expira sem receber todas as assinaturas esperadas).

4.2.3 Tabela de requisitos funcionais

A Tabela 1 reúne, de forma sistematizada, os requisitos funcionais discutidos nas subseções anteriores, com identificador e descrição de cada um, servindo como referência para a implementação e a verificação do PdD.

Tabela 1 – Requisitos Funcionais do Preparador de Documentos.

Identificador (ID)	Descrição
RF1	O sistema deve permitir ao usuário o cadastro de signatários ao documento, incluindo dados como nome, e-mail e/ou número de telefone, descrição e comprometimento em relação ao documento.
RF2	O sistema deve permitir que o usuário que prepara o documento também realize sua própria assinatura sobre ele.
RF3	O sistema deve permitir ao usuário o preparo de documentos para serem assinados sem incluir sua própria assinatura.
RF4	O sistema deve permitir ao usuário acompanhar o <i>status</i> das assinaturas dos documentos preparados.
RF5	O sistema deve permitir ao usuário o envio de notificações de assinatura requerida a signatários por e-mail ao preparar o documento.
RF6	O sistema deve permitir ao usuário o envio de notificações de assinatura requerida a signatários por meio do WhatsApp ao preparar o documento.
RF7	O sistema deve permitir ao usuário que prepara o documento o recebimento de notificação quando todos signatários de seu documento tiverem realizado suas assinaturas.
RF8	O sistema deve permitir ao usuário que prepara o documento especificar uma data limite para recebimento das assinaturas.
RF9	O sistema deve permitir ao usuário que prepara o documento definir a posição do artefato visual que será associado a cada assinatura no documento PDF no fluxo de preparo com signatários especificados.
RF10	O sistema deve tratar e apresentar visualmente ao usuário estados intermediários do processo de coleta de assinaturas, como "recebendo assinaturas", "incompleto" e "completo".
RF11	O sistema deve permitir ao usuário preparar e compartilhar documentos por meio de um link compartilhado de acesso, isto é, sem a necessidade de especificar previamente signatários esperados, mantendo um mecanismo claro de obtenção e compartilhamento desse link.
RF12	O sistema deve permitir ao usuário que é signatário especificado em um documento a realização de sua assinatura com apenas um clique após a autenticação.
RF13	O sistema deve permitir ao usuário que prepara o documento especificar uma mensagem a ser enviada juntamente à notificação ao signatário.

Identificador (ID)	Descrição
RF14	O sistema deve permitir ao usuário que, ao preparar o documento, realiza sua própria assinatura e não espera nenhum outro signatário o download do documento já assinado com um só clique após a finalização do processo de preparo.
RF15	O sistema deve disponibilizar ao usuário que prepara o documento um link que pode ser compartilhado por meios externos ao sistema para acesso ao documento ao final do fluxo de preparo.
RF16	O sistema deve permitir ao usuário que prepara o documento o download do documento preparado em qualquer momento do processo de recebimento de assinaturas após o documento ter sido preparado, mas com aviso prévio sobre a situação do documento caso não tenha sido concluído, isto é, recebido todas as assinaturas esperadas no fluxo com signatários especificados ou tenha chegado o prazo limite no fluxo de compartilhamento por link.
RF17	O sistema deve notificar o usuário que prepara o documento quando um documento com signatários especificados que preparou chegou no prazo limite e não recebeu todas assinaturas.
RF18	O sistema deve permitir ao usuário que prepara o documento inserir uma descrição e o comprometimento de cada signatário em relação ao documento no momento de seu cadastro.
RF19	O sistema deve permitir ao usuário que prepara o documento realizar sua assinatura com apenas um clique, sem necessidade de posicionar artefato visual, após inserir o documento no sistema.
RF20	O sistema deve permitir ao usuário que prepara o documento obter e compartilhar por link um documento já inserido/preparado no sistema, mesmo que ele não tenha sido inicialmente configurado para compartilhamento por link ao final do fluxo de preparo.
RF21	O sistema deve permitir ao usuário que prepara o documento adicionar-se como signatário esperado e optar por não realizar sua própria assinatura ao finalizar o preparo, permitindo assiná-lo posteriormente.

Fonte: elaborado pelo autor.

4.3 FLUXO DE TELAS PROPOSTO EM TRABALHO ANTERIOR

Lima (2023) criou interfaces para o preparador e um protótipo navegável³ com base nos objetivos de uso coletados em seu trabalho. Foram feitas no Figma, uma ferramenta de prototipação, e não constituem uma aplicação *web* integrada ao ADES. No presente trabalho, essas

³ <https://www.figma.com/proto/C4GEKR0JcTj55scYMjeSuP/Assinador-2023>

telas servem como referência de design para a implementação do fluxo de interação do PdD no PAS, descrita no Capítulo 6, com as adaptações necessárias aos novos requisitos levantados e ao uso em *desktop*.

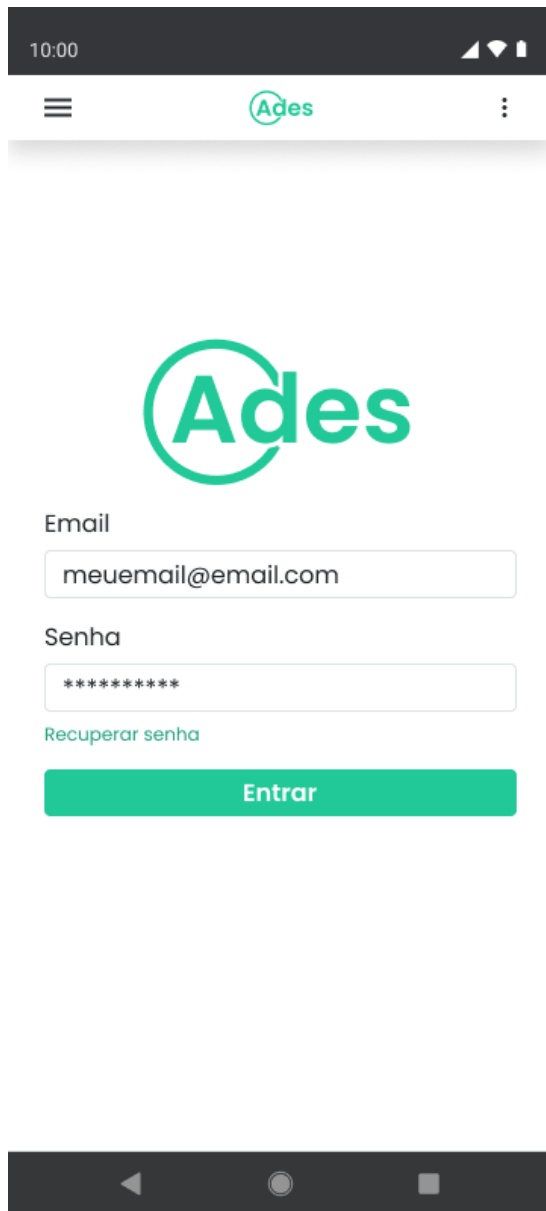
No fluxo sugerido, espera-se que o usuário que irá preparar o documento inicialmente autentique-se (Figura 9). Então, é recebido com uma tela que permite preparar um novo documento ou ver documentos já preparados por ele (Figura 10). Ao abrir um documento que está coletando assinaturas, é possível visualizá-lo e também ver detalhes como assinaturas já recebidas e pendentes (Figura 11). Caso prefira inserir um novo documento no sistema, deverá prepará-lo para receber assinaturas, especificando detalhes como quem deve assinar e se o próprio usuário também assinará o documento (Figura 12). A adição de cada signatário conta com a especificação do seu e-mail para identificá-lo (Figura 13). Assim que quem prepara finaliza o processo, resta aguardar pela coleta das assinaturas (Figura 15).

4.3.1 Protótipo interno desenvolvido no LabSEC

Entre o design de Lima (2023) e a implementação integrada realizada neste trabalho, internamente no LabSEC um membro da equipe desenvolveu, no início de 2025, um protótipo navegável desse fluxo utilizando *React* com *Javascript*, *HTML* e *CSS*.⁴ Esse protótipo não se integrava ao sistema do projeto ADES e, portanto, não permitia a inserção real de documentos no sistema. Ainda assim, foi útil como forma dinâmica de testar e discutir o fluxo de interação e código-fonte com estrutura de referência dos componentes do sistema antes da codificação definitiva no PAS.

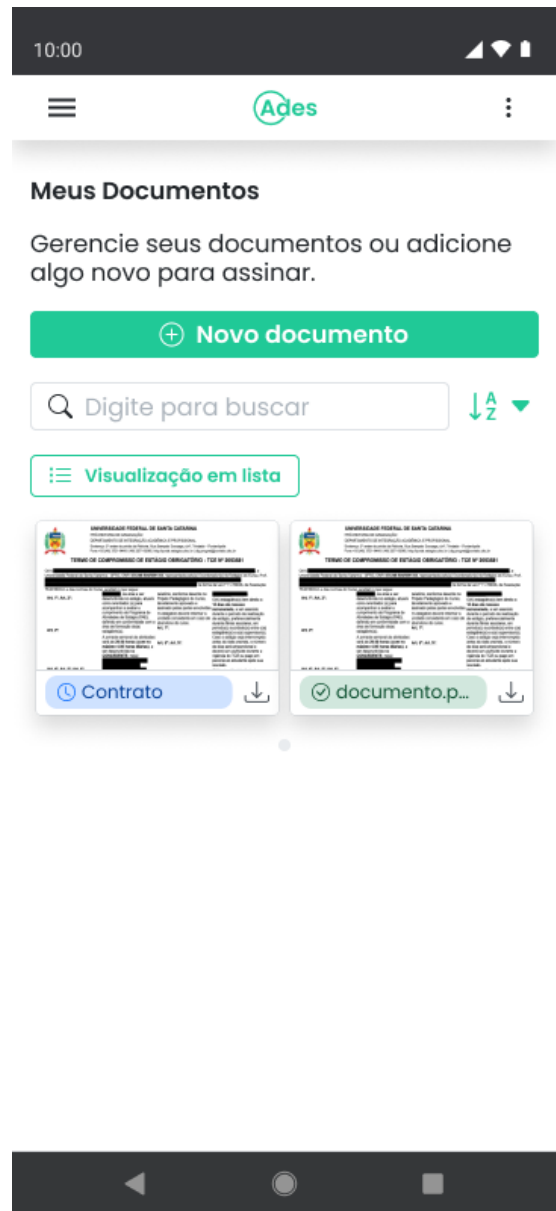
⁴ Protótipo implementado por Brendon Vicente Rocha Silva, ex-membro do LabSEC, no início de 2025.

Figura 9 – Tela exemplo de autenticação.



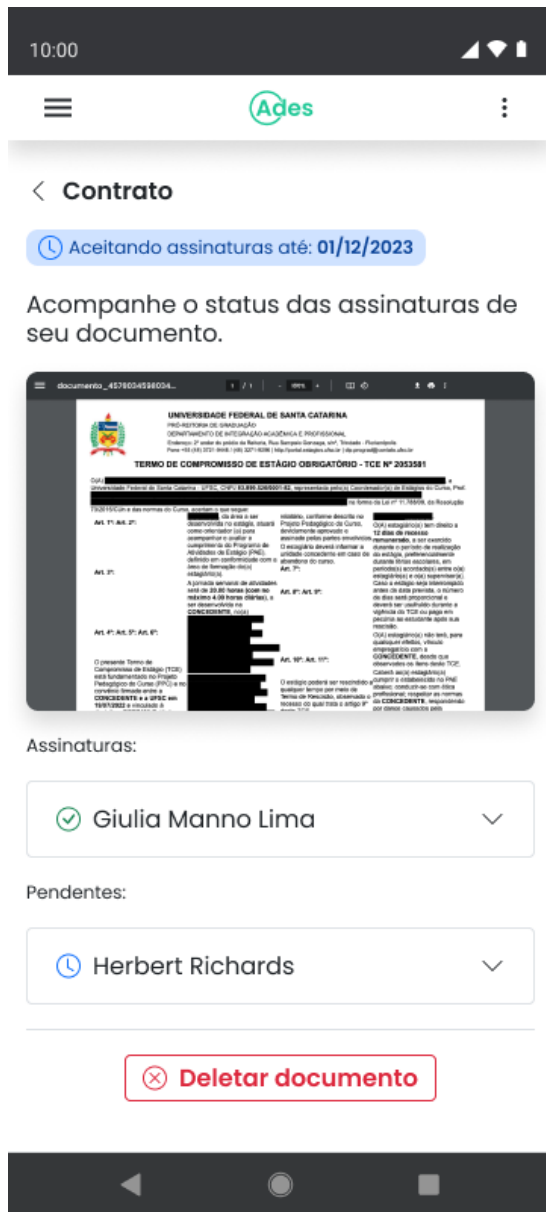
Fonte: Lima (2023).

Figura 10 – Tela exemplo da lista de documentos.



Fonte: Lima (2023).

Figura 11 – Tela exemplo de documento que foi preparado.



Fonte: Lima (2023).

Figura 12 – Tela exemplo de especificação de signatários.



Fonte: Lima (2023).

Figura 13 – Tela exemplo de adição de signatário no processo de preparo.

10:00

Ades

< Novo documento

Adicionar signatário

Nome*

Herbert Richards

E-mail*

versbrsileira@hrbtrchrds.com

Comprometimento

Contratado

Insira a relação do signatário com o documento.

Salvar

Voltar Continuar

Fonte: Lima (2023).

Figura 14 – Tela exemplo da finalização do processo de preparo.

10:00

Ades

< Novo documento

Revise as opções de compartilhamento.

Receber assinaturas até: 01/12/2023

Após encerramento do recebimento das assinaturas, os documentos ficam armazenados no sistema por até 30 dias.

☒ Avisar signatários por e-mail

Mensagem

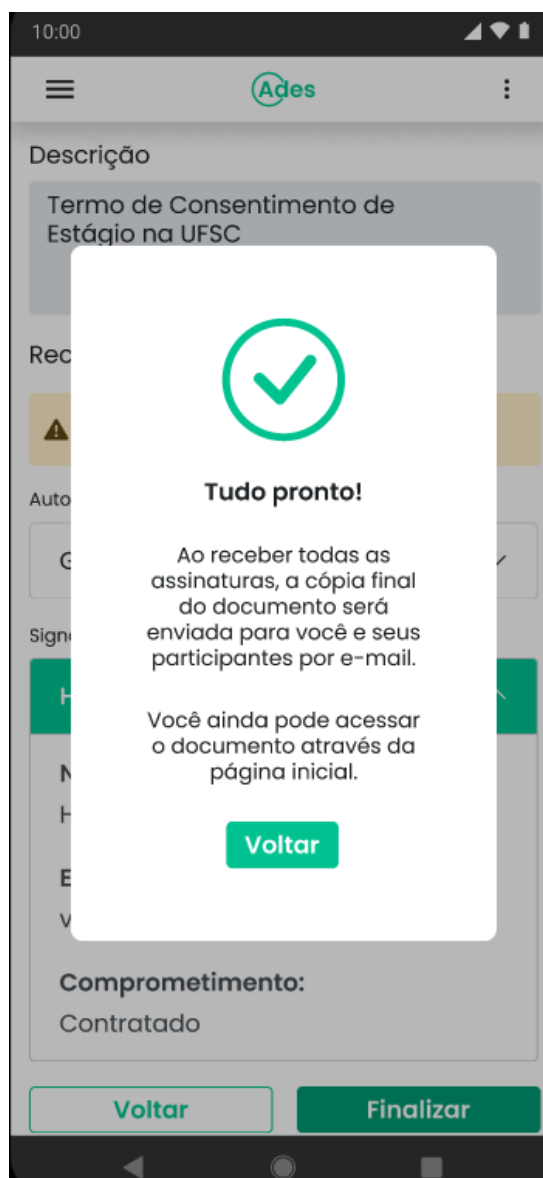
Por favor, assine meu documento através do link: link.bit.ly/3574378

0/200

Voltar Continuar

Fonte: Lima (2023).

Figura 15 – Tela exemplo da finalização do processo de preparo.



Fonte: Lima (2023).

4.3.2 Atendimento dos requisitos no fluxo sugerido

As telas não só suprem boa parte dos objetivos de uso do Preparador mas também propõem funcionalidades não obrigatórias que visam melhorar a experiência do usuário em relação ao ciclo de vida dos documentos, como filtros na lista de documentos, prazo para assinaturas, indicadores visuais de progresso no processo de preparo, entre outros. Esta seção discute o atendimento dos requisitos no fluxo sugerido, dizendo se é coberto, parcialmente coberto ou não contemplado, e sintetizando na tabela 2.

O RF1 é atendido pela tela que permite especificar signatários (Figura 11) e seu nome, e-mail e comprometimento (Figura 13). O RF2 e RF3 são atendidos pela tela de especificação de signatários (Figura 12). O RF4 é atendido pela tela que mostra documento que foi

preparado, onde há uma lista de assinaturas recebidas e pendentes (Figura 11). Já o RF5 e RF13 são visíveis na tela de revisão das opções de compartilhamento, onde é possível especificar que os signatários devem ser notificados e receber um aviso por e-mail, acompanhado de mensagem customizada (Figura 14). O RF18 é atendido ao ser possível especificar o comprometimento de cada signatário (Figura 13) e uma mensagem (Figura 15). O protótipo de Lima (2023) contempla, ainda, a apresentação de estados intermediários do documento e fluxos de compartilhamento por link (RF10, RF11, RF15 e RF20), embora essas telas não tenham sido reproduzidas nas figuras desta análise. Por fim, o RF7 não é visível pelas telas fornecidas, mas pode ser implementado na lógica do lado do servidor, para que o usuário, por exemplo, receba um e-mail quando todas assinaturas tiverem sido coletadas.

4.3.3 Limitações do fluxo de telas proposto

Embora as interfaces propostas por Lima (2023) atendam a uma parcela significativa dos requisitos funcionais e forneçam um direcionamento consistente para a implementação do PdD, o levantamento de novos requisitos e casos de uso em comparação aos que foram utilizados no seu trabalho trouxeram algumas limitações à utilização do seu design da maneira apresentada.

Primeiramente, o design proposto no trabalho analisado foi concebido com foco primário em dispositivos móveis, o que impõe a necessidade de adaptação para uso em ambientes *desktop*. O protótipo interno desenvolvido no LabSEC consta com telas *desktop*. Além disso, não é sugerido um design específico para o conteúdo dos e-mails enviados pelo sistema, ponto associado aos requisitos RF5, RF6, RF7, RF13 e RF17, uma vez que o envio de notificações pressupõe não apenas a existência do mecanismo de disparo, mas também a definição do conteúdo comunicado ao signatário, incluindo instruções, contexto e eventuais informações complementares.

Observa-se também a inexistência de uma proposta de interface que permita ao usuário escolher a posição em que será inserido o artefato visual correspondente a cada assinatura recebida dos signatários esperados (RF9). Tampouco há proposta de design para os avisos exibidos quando o usuário que preparou tenta realizar o *download* antes da conclusão do documento (RF16).

Percebe-se também que a tela proposta quando o usuário que prepara o documento também o assina e não há outros signatários esperados não conta com download facilitado assim que o processo de preparo é concluído (RF14), exigindo retorno à tela de lista de documentos para realizá-lo. Quando o usuário quer realizar sua própria assinatura, ele consegue fazê-lo inserindo um documento e avançando por todas as etapas do processo de preparo, o que atenderia parcialmente o RF19, porém a grande quantidade de cliques não atende a essa exigência trazida pelo requisito. Quando o usuário se adiciona como signatário esperado, o sistema já realiza sua assinatura ao concluir o processo de preparo, o que conflita com o requisito RF21, que estabelece a possibilidade de o usuário optar por não realizar sua assinatura de imediato.

Tabela 2 – Atendimento dos requisitos funcionais pelo fluxo de telas proposto em trabalho anterior.

Requisito (ID)	Atendido?	Como é atendido
RF1	Sim	Tela permite especificar signatários com nome, e-mail e comprometimento (Figura 11; Figura 13).
RF2	Sim	Tela de especificação de signatários permite que o próprio usuário seja incluído como signatário (Figura 12).
RF3	Sim	A mesma tela de especificação de signatários permite não incluir a própria assinatura no fluxo (Figura 12).
RF4	Sim	Tela de documento preparado exibe lista de assinaturas recebidas e pendentes (Figura 11).
RF5	Parcialmente	Tela de revisão permite ativar notificações por e-mail aos signatários (Figura 14), mas não há design do conteúdo do e-mail.
RF6	Não	Não há proposta de notificação de pedido de assinatura por WhatsApp.
RF7	Parcialmente	Não visível nas telas propostas; pode ser implementado na lógica do servidor, mas sem design definido.
RF8	Sim	Prazo para recebimento de assinaturas está presente entre as funcionalidades propostas.
RF9	Não	Não há proposta de interface para posicionamento do artefato visual de cada signatário.
RF10	Sim	Previsto no protótipo de Lima (2023).
RF11	Sim	Previsto no protótipo de Lima (2023).
RF12	Não	Não abordado no fluxo proposto.
RF13	Parcialmente	Mensagem customizada prevista na tela de revisão de compartilhamento (Figura 14), mas sem design do conteúdo do e-mail.
RF14	Não	Tela de conclusão do preparo não disponibiliza download facilitado quando o preparador é o único signatário esperado.
RF15	Sim	Previsto no protótipo de Lima (2023).
RF16	Não	Não há design para aviso de download de documento com assinaturas incompletas.
RF17	Não	Não contemplado.
RF18	Sim	Possível especificar comprometimento de cada signatário (Figura 13) e mensagem adicional (Figura 15).

Requisito (ID)	Atendido?	Como é atendido
RF19	Parcialmente	Possível assinar avançando por todas as etapas do preparo, mas a quantidade de cliques não atende à exigência do requisito.
RF20	Sim	Previsto no protótipo de Lima (2023).
RF21	Não	O fluxo já realiza a assinatura do usuário ao concluir o preparo quando ele se adiciona como signatário, sem possibilidade de adiamento.

Fonte: elaborado pelo autor.

5 PROPOSTA

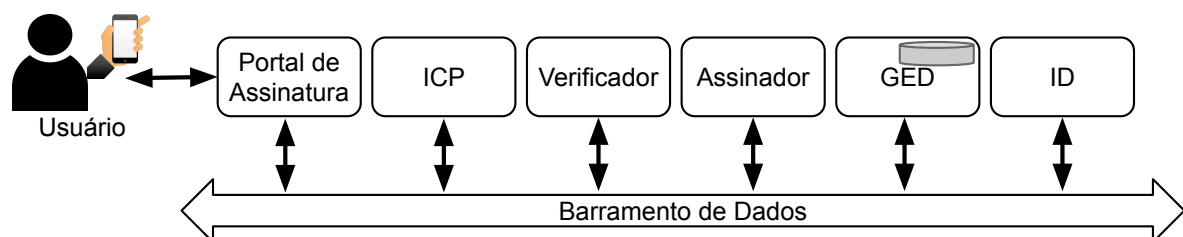
Após o levantamento de requisitos e a análise do fluxo de telas proposto em trabalho anterior, este capítulo apresenta a proposta de inserção do PdD no ecossistema do ADES. Contextualiza-se a arquitetura de microserviços do projeto, os serviços impactados, os fluxos de processo modelados em BPMN e o papel de cada serviço no preparo e na assinatura de documentos. A implementação concreta no código é descrita no Capítulo 6; as evidências de uso na aplicação, no Capítulo 7.

5.1 ARQUITETURA DO PROJETO ADES E SERVIÇOS IMPACTADOS

A arquitetura adotada pelo sistema ADES baseia-se no padrão de microserviços, onde ocorre a divisão da aplicação em diferentes serviços, cada um com responsabilidades específicas (Perottoni et al., 2023).

A Figura 16 mostra os serviços que constituem a arquitetura: o PAS, que é o meio de interação do usuário com o sistema; ICP é o serviço que se comunica com a ICP para emitir os certificados OTC usados nas assinaturas digitais; Verificador é responsável por mostrar detalhes de assinaturas em documentos, como quantidade, detalhes do signatário e sistema assinador e se são válidas ou não; Assinador realiza a assinatura digital em documentos do tipo PDF; GED gerencia o ciclo de vida de documentos inseridos no sistema a partir do PAS; e ID, responsável por realizar a autenticação e coletar atributos do usuário, conecta-se com o IdP para autenticar usuários e coletar seus atributos para emissão da assinatura (Perottoni et al., 2023).

Figura 16 – Arquitetura do sistema ADES.



Fonte: Perottoni et al. (2023)

A implementação do PdD contemplará alterações no GED e precisará lidar com os novos estados do processo de preparo de documento, e no PAS, que fornece as interfaces no navegador para o usuário interagir com o sistema e que receberá as novas telas e funcionalidades. No envio de notificações a signatários esperados, o GED gera links individualizados com token

único por convidado; a validação desse token e a autenticação automática do signatário ficam a cargo do microserviço ID, já presente na arquitetura do ADES (Figura 16).

5.1.1 Portal de Assinatura e Gerenciador Eletrônico de Documentos

A aplicação *web* desenvolvida com *React* e *Typescript* será responsável por fornecer as interfaces no navegador para o usuário interagir com o PdD. Por meio dela, o usuário será capaz de realizar o preparo de documentos, gerenciar seus documentos preparados e acompanhar o andamento do processo de preparo.

O GED irá armazenar documentos preparados junto a informações sobre seus signatários esperados, completude, expiração, recebimento de novas tentativas de assinatura, entre outros. Além disso, será o responsável por notificar os usuários envolvidos no processo de assinatura do documento, por e-mail.

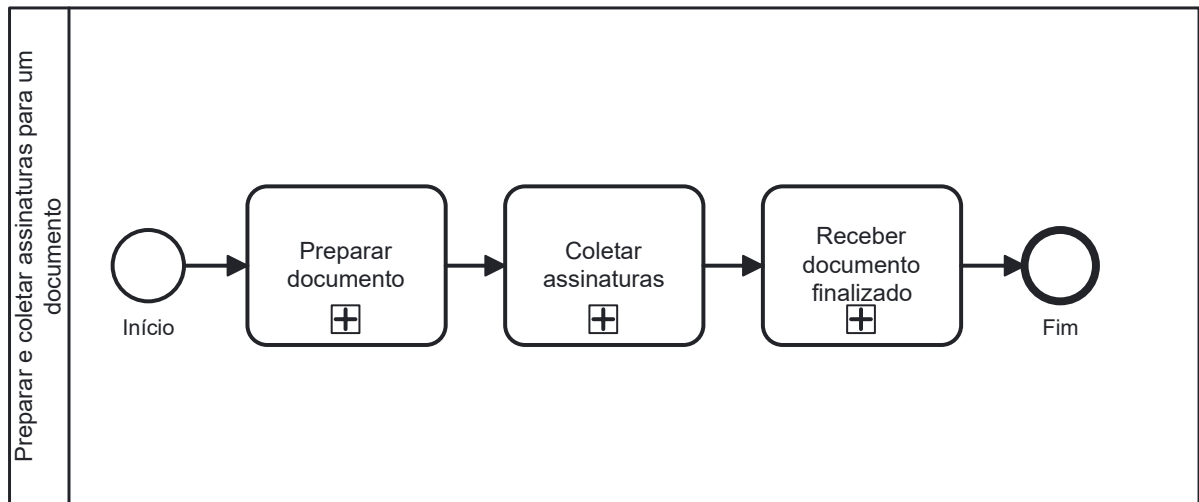
5.2 FLUXOS DE PROCESSO DO PREPARADOR DE DOCUMENTOS

Os fluxos a seguir modelam o comportamento proposto para o PdD em notação BPMN. Especificam etapas, atores e decisões que orientam a implementação.

5.2.1 Fluxo principal: preparo, coleta e recebimento

O fluxo principal contempla o preparo do documento, a coleta de assinaturas e o recebimento do documento finalizado por quem preparou. A Figura 17 sintetiza o processo em três partes: (i) preparar o documento, inserindo-o no sistema e especificando signatários; (ii) coletar assinaturas, com notificação aos signatários esperados; e (iii) receber o documento finalizado, com possibilidade de download.

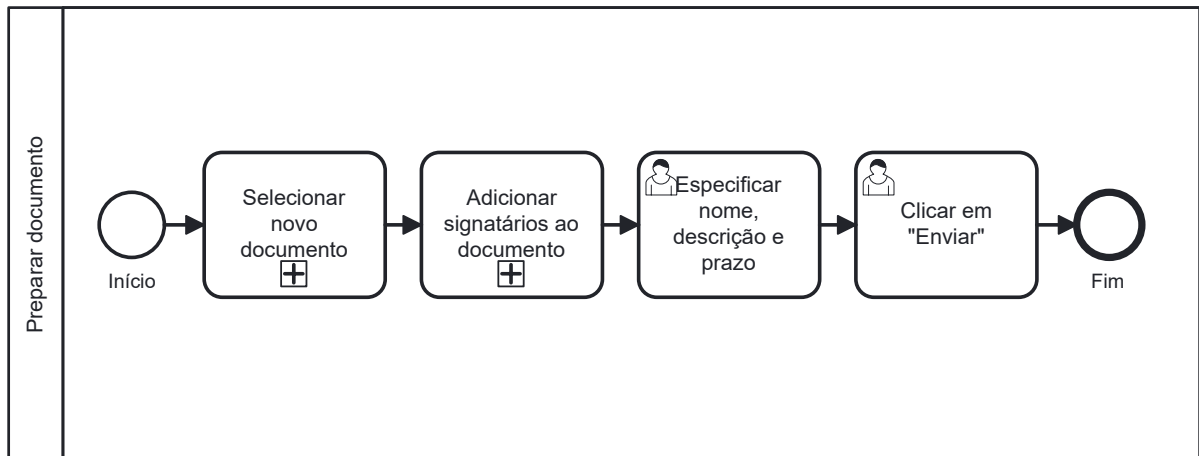
Figura 17 – Processo proposto de preparo de documentos e coleta de assinaturas.



Fonte: elaborado pelo autor.

O preparo é detalhado na Figura 18: seleção do documento, especificação dos signatários esperados, informações opcionais (nome, descrição, prazo limite) e envio.

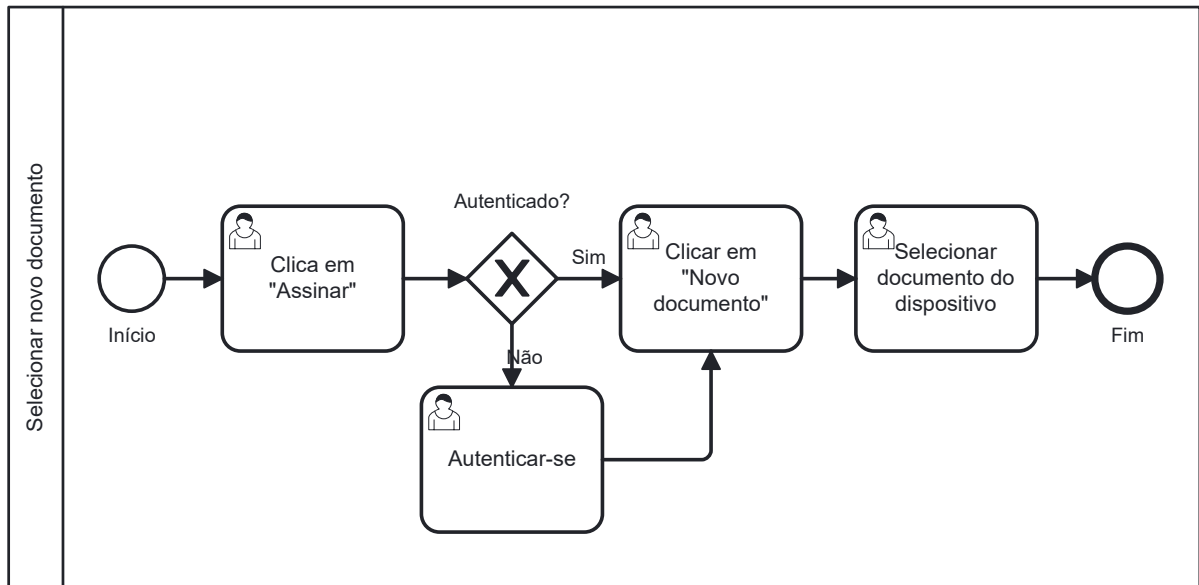
Figura 18 – Etapas propostas do preparo de um documento no Preparador de Documentos.



Fonte: elaborado pelo autor.

A seleção de um novo documento segue o fluxo da Figura 19: acesso ao PAS, autenticação quando necessário, consulta à lista de documentos preparados, escolha de um arquivo PDF e abertura da tela de preparo.

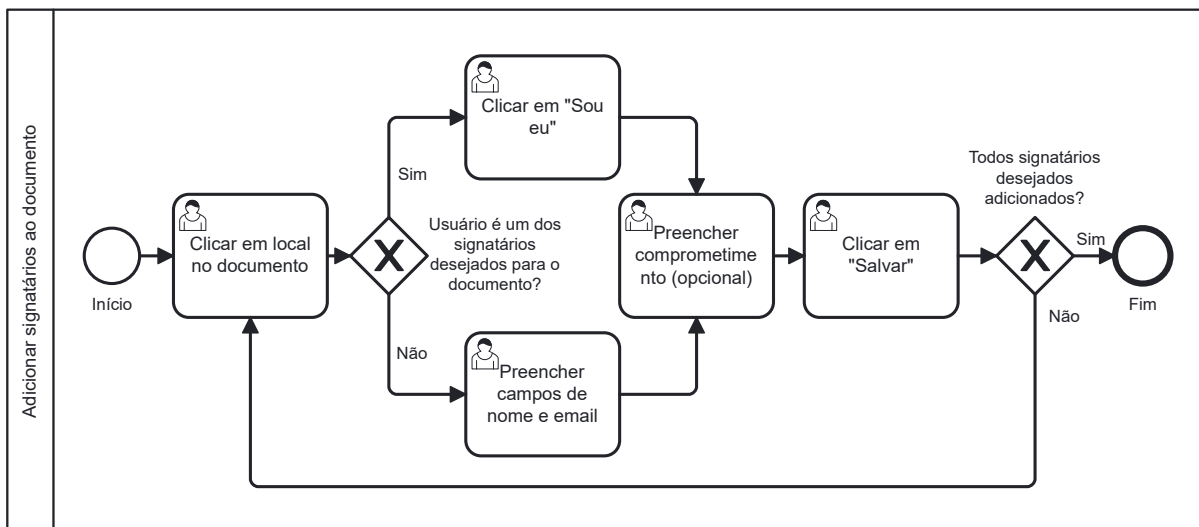
Figura 19 – Fluxo proposto para selecionar um novo documento.



Fonte: elaborado pelo autor.

A inclusão de signatários e o posicionamento do artefato visual no PDF seguem a Figura 20: o usuário indica no documento onde cada assinatura será representada, cadastra dados do signatário (nome, e-mail ou número de telefone, comprometimento) e pode incluir-se como signatário esperado. Ao concluir o preparo, o sistema processa o envio e, quando quem prepara consta entre os signatários esperados, pode registrar sua assinatura nesse momento.

Figura 20 – Fluxo proposto de inclusão de signatário e posicionamento no PDF.

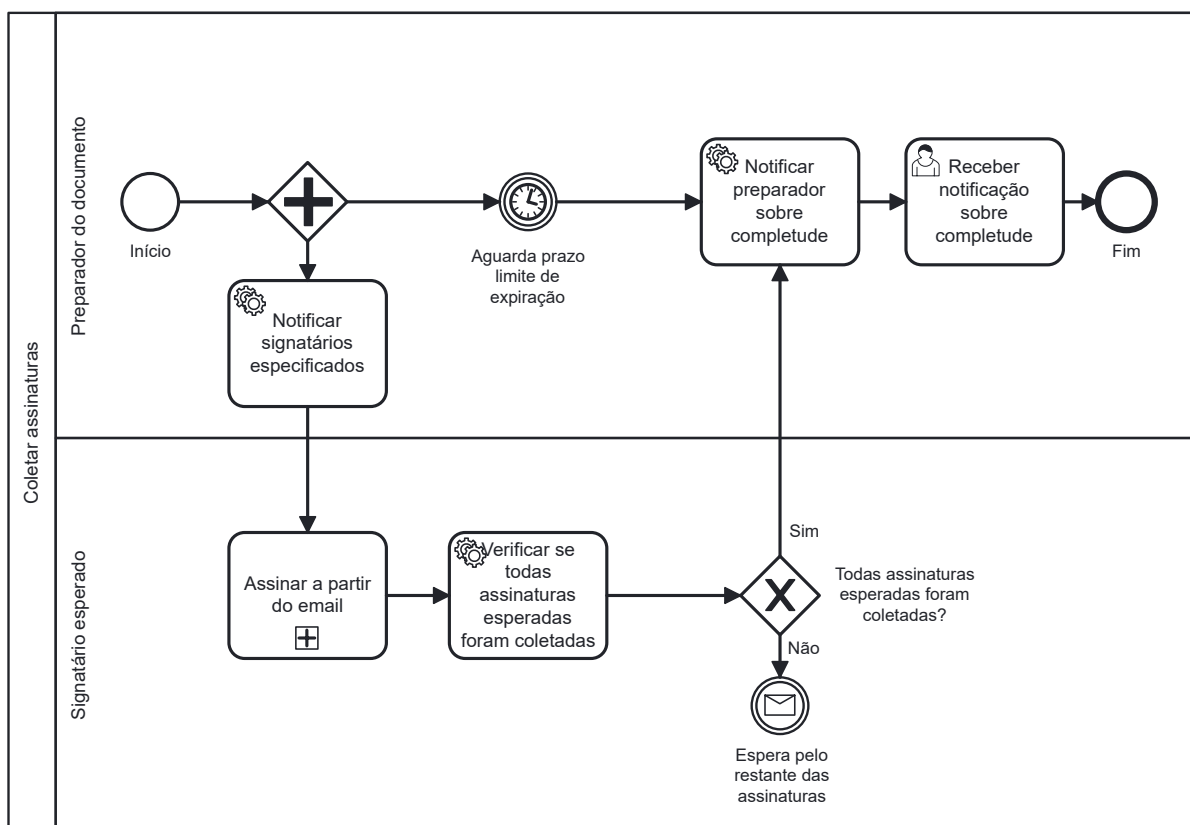


Fonte: elaborado pelo autor.

Após o preparo, inicia-se a coleta de assinaturas (Figura 21). Os signatários esperados podem ser notificados automaticamente; o documento conclui-se ao receber todas as assinaturas

cadastradas ou ao expirar o prazo, com aviso a quem preparou quando a coleta fica incompleta. Quando o lote é compartilhado por link (is_public), qualquer usuário autenticado com o endereço pode assinar; se o e-mail autenticado corresponder a um signatário esperado, a assinatura utiliza o artefato visual predefinido no preparo. Nos trâmites abertos, sem signatários esperados, a conclusão ocorre apenas ao expirar o prazo.

Figura 21 – Processo proposto de coleta de assinaturas.

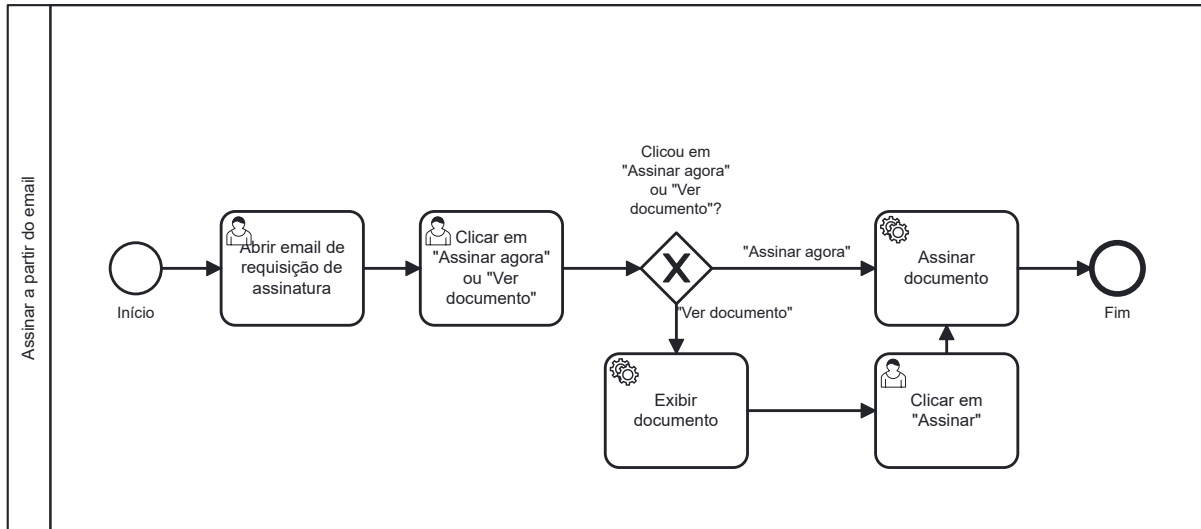


Fonte: elaborado pelo autor.

O signatário esperado pode assinar a partir do e-mail recebido (Figura 22), com opção de assinar de imediato ou visualizar o documento no sistema antes.

Cada notificação enviada a um signatário esperado, por e-mail ou por WhatsApp, inclui um link individualizado que incorpora um token único, associado ao convite daquele signatário para aquele lote de documento. Ao abrir o endereço no navegador, o microserviço ID valida o token, identifica o signatário correspondente e estabelece sua sessão autenticada no sistema, dispensando o fluxo usual de login no IdP. Com isso, o signatário pode assinar com um clique a partir da mensagem recebida (RF12), e o sistema vincula a assinatura ao registro correto em `batch\_signers`, aplicando o artefato visual nas coordenadas definidas no preparo. Do ponto de vista de segurança, essa autenticação implícita parte da premissa de que o acesso ao canal pessoal de comunicação, a caixa de e-mail ou o aplicativo de mensagens no celular, constitui um fator de posse sobre dados pessoais do destinatário.

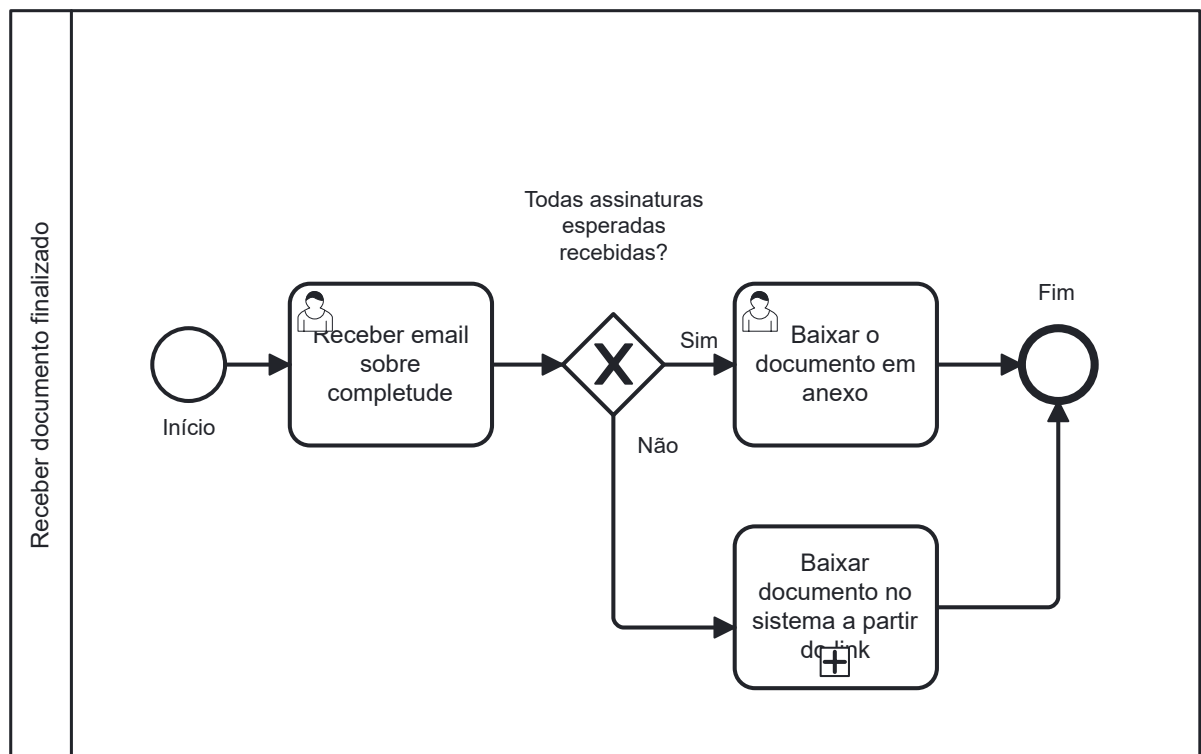
Figura 22 – Processo proposto de assinatura pelo signatário esperado a partir do e-mail.



Fonte: elaborado pelo autor.

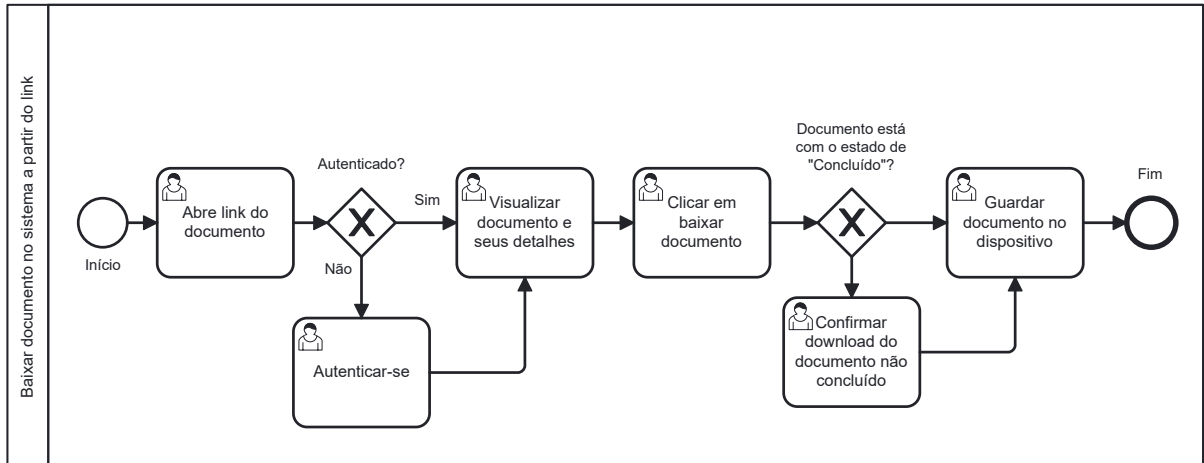
Ao final da coleta, quem preparou é notificado por e-mail com o documento em anexo e link para visualização no sistema (Figura 23). Alternativamente, pode autenticar-se e realizar o download pelo *site* (Figura 24).

Figura 23 – Processo proposto de recebimento do documento finalizado.



Fonte: elaborado pelo autor.

Figura 24 – Processo proposto de download do documento a partir do link no *site*.



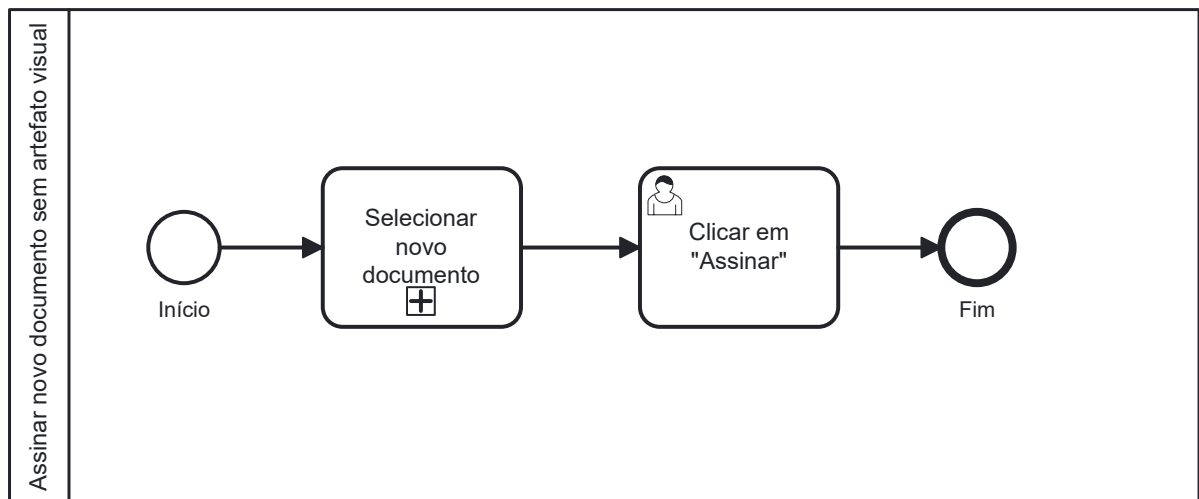
Fonte: elaborado pelo autor.

Além do fluxo principal, o PdD prevê variações de uso, modeladas a seguir.

5.2.2 Assinatura com um clique sem artefato visual

Quando o usuário é o único signatário esperado, após inserir o documento no sistema, o fluxo proposto permite assiná-lo com um clique no botão de assinatura, sem posicionar artefato visual (Figura 25).

Figura 25 – Fluxo proposto de assinatura com um clique sem posicionamento de artefato visual.



Fonte: elaborado pelo autor.

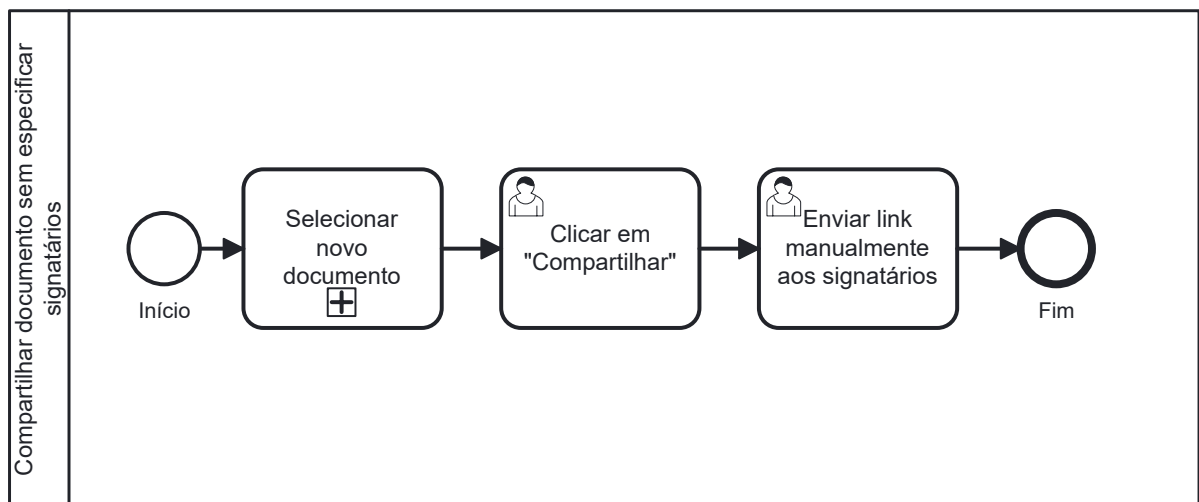
5.2.3 Compartilhamento por link sem signatários esperados

O preparo pode dispensar signatários previamente definidos e gerar um link de acesso compartilhável; nesse trâmite aberto, qualquer usuário autenticado que possua o endereço pode assinar, sem posicionamento prévio de artefatos visuais, e o documento finaliza-se ao atingir o prazo limite (Figura 26).

5.2.4 Compartilhamento por link com signatários esperados

O compartilhamento por link não se restringe ao trâmite aberto: é possível cadastrar signatários esperados, definir prazos, posicionar artefatos visuais e, ainda assim, disponibilizar um link único para divulgação externa. Nesse cenário combinado, o link funciona como via de acesso ampliada: quem o recebe precisa autenticar-se no IdP antes de visualizar ou assinar. Se o e-mail autenticado corresponder a um dos signatários cadastrados em `batch\_signers`, o sistema associa a assinatura a esse registro e aplica as coordenadas definidas no preparo; caso contrário, registra uma assinatura adicional de quem não constava na lista de esperados. Assim, o preparador mantém o acompanhamento das assinaturas pendentes entre os convidados nomeados, enquanto o link viabiliza o acesso a quem não recebeu ou não viu a notificação automática.

Figura 26 – Fluxo proposto de compartilhamento por link sem especificar signatários.

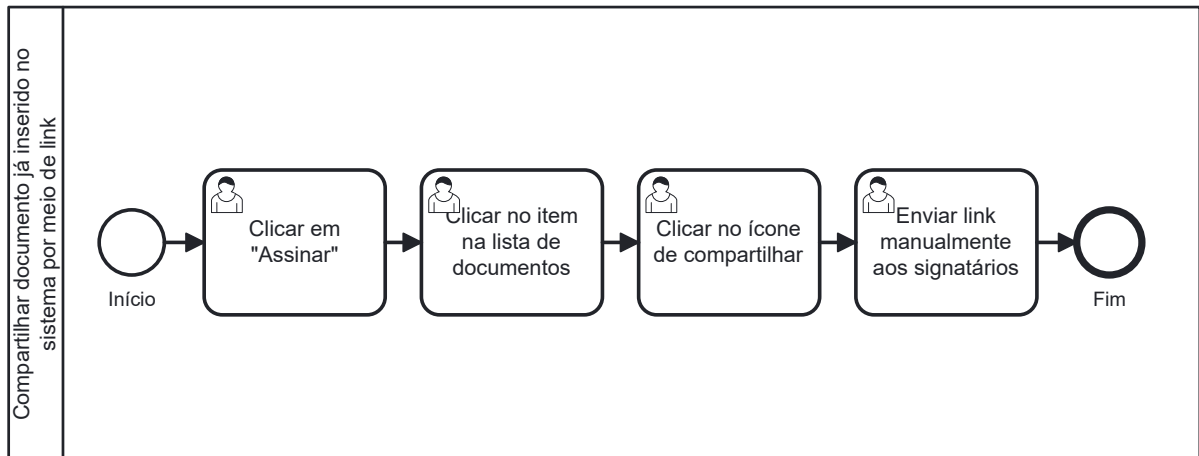


Fonte: elaborado pelo autor.

5.2.5 Compartilhamento de documento já inserido no sistema

Documentos já preparados podem passar a ser compartilhados por link a partir da lista de documentos (Figura 27).

Figura 27 – Processo proposto de compartilhamento por link de documento já inserido no sistema.



Fonte: elaborado pelo autor.

5.2.6 Adiar a própria assinatura

Durante o preparo, quem adiciona-se como signatário esperado pode optar por não assinar ao concluir o envio, realizando a assinatura posteriormente na página de detalhes do documento.

5.2.7 Monitoramento e estados do documento

Documentos preparados devem exibir estados como concluído, pendente e incompleto na lista de documentos, permitindo acompanhar o andamento da coleta de assinaturas e situações em que o prazo expira sem todas as assinaturas esperadas.

6 IMPLEMENTAÇÃO

Com base na proposta apresentada no Capítulo 5, este capítulo descreve a implementação realizada do PdD no ecossistema do ADES. São detalhadas as alterações no PAS e no GED.

6.1 ADIÇÕES AO PORTAL DE ASSINATURA

O PdD é composto por um conjunto de arquivos e pastas que organizam o código-fonte da aplicação. Foi criada uma subpasta `preparador` para organizar os arquivos do código-fonte referentes a essa nova funcionalidade. A Tabela 3 resume a organização dessa estrutura no PAS, onde se concentram os componentes desenvolvidos em *React* e *Typescript*.

Tabela 3 – Organização das pastas do Preparador de Documentos no código-fonte do Portal de Assinatura.

Pasta ou arquivo	Responsabilidade
<code>pages/</code>	Telas de navegação do módulo (listagem, preparo, detalhes do documento e erros).
<code>components/</code>	Componentes <i>React</i> reutilizáveis da interface (preparo, listagem, visualização, signatários e validações).
<code>contexts/</code>	Compartilhamento de estado entre componentes da interface.
<code>types/</code>	Definição dos tipos do módulo em <i>TypeScript</i> .
<code>utils/</code>	Funções auxiliares (comunicação com o <i>backend</i> , datas e validações).
<code>styles/</code>	Estilos visuais específicos do módulo.
<code>PreparadorDeDocumentos.tsx</code>	Orquestração do módulo, <i>upload</i> de arquivos e roteamento entre páginas.

Fonte: elaborado pelo autor.

Na pasta `pages` encontram-se as novas páginas do sistema, como a `DashboardPage.tsx` que exibe a lista de documentos preparados pelo usuário, a `NewDocumentPage.tsx` que fornece o fluxo de preparo de novo documento e a `DocumentInfoPage.tsx` para visualização e informações detalhadas do documento preparado. Há ainda a `ErrorPage.tsx` para tratamento de erros inesperados.

Já a pasta `components` contém os componentes *React* reutilizáveis da interface. Dentro dela, a subpasta `NewDocument` concentra o fluxo de preparo de novos documentos no componente `PrepareDocument.tsx`, que reúne em uma única tela o posicionamento dos artefatos visuais no PDF, a inclusão e edição de signatários, a definição de prazo e a revisão dos dados antes do envio. Essa pasta também inclui `documentSteps.ts`, que descreve os passos do fluxo, e `useFinalizeNewDocument.ts`, responsável por finalizar o preparo e enviar o documento ao *backend*. A subpasta `DocumentList` contém componentes para exibição e gerenciamento da lista de documentos, incluindo filtros, ordenação e paginação. Outros componentes incluem `PDFViewer.tsx` para visualização de documentos PDF, `SignerCard.tsx` para exibição de

informações de signatários e componentes de modais para tratamento de erros de validação de arquivos.

A pasta `contexts` armazena os contextos *React* utilizados para compartilhar estado entre componentes, como `PreparadorContexts.tsx` que gerencia os dados do documento sendo preparado e o estado do visualizador de PDF, e `DocumentBufferContext.tsx` para gerenciamento de *buffer* de documentos.

Na pasta `types` definem-se tipos *TypeScript* utilizados no PdD, centralizadas no arquivo `PreparadorTypes.ts`, que define interfaces para documentos, signatários, estados de documentos e outros tipos relacionados.

Em `utils` agrupam-se funções utilitárias que realizam operações auxiliares, como comunicação com a *API* do *backend* para obter listas de documentos (`getDocumentList.ts`), informações de documentos (`getDocumentInfo.ts`), *status* de documentos (`getDocumentStatus.ts`), *upload* de documentos (`uploadDocument.ts`), manipulação de datas (`dateUtils.ts`, `parseDocumentDates.ts`) e validações diversas.

A pasta `styles` contém os arquivos de estilização *SCSS* específicos do PdD, como `index.scss` com estilos gerais e `sectionHeader.scss` para estilização de cabeçalhos de seção.

Por fim, o arquivo `PreparadorDeDocumentos.tsx` é o componente principal que gerencia o fluxo geral do preparador, incluindo o tratamento de *upload* de arquivos e a estrutura de roteamento das páginas.

6.2 MUDANÇAS NO GERENCIADOR ELETRÔNICO DE DOCUMENTOS

Além das alterações no PAS, o PdD exige mudanças no GED, tanto no esquema de dados quanto nas rotas e regras de negócio do serviço.

6.2.1 Ajustes no banco de dados

Antes do PdD, o ADES utilizava 3 tabelas em um banco de dados para armazenar dados relacionados a documentos e permissões de acesso a eles.

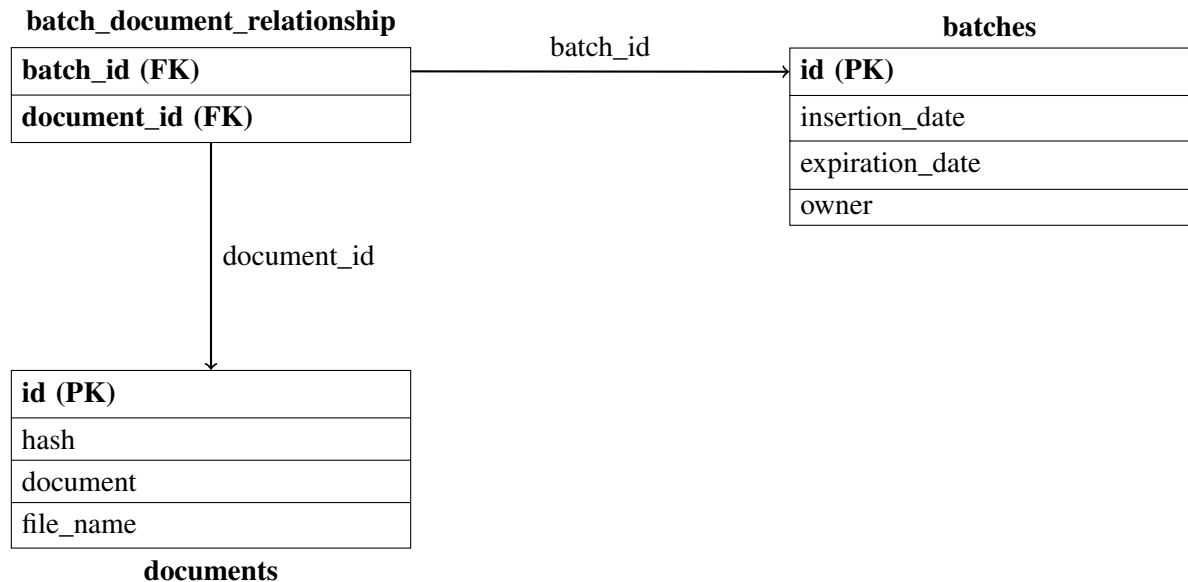
A tabela `documents` contava com colunas com dados do documento como identificador único, resumo criptográfico do conteúdo, conteúdo e nome do arquivo. Essa tabela é utilizada para armazenar os documentos inseridos no sistema.

Quando o usuário do sistema realiza uma assinatura ou verificação, a rota do navegador é alterada para mostrar um identificador único referente ao documento inserido no sistema, de forma que essa mesma rota pode ser reutilizada para abrir o documento novamente caso o usuário saia do *site*. Entretanto, esse identificador não é o mesmo que o do documento, e sim de um lote, uma abstração criada junto à inserção do documento. Assim, a tabela `batches` que guarda os lotes conta com identificador único, data de inserção, data de expiração e proprietário.

A associação entre lote e documento é feita na tabela `batch_document_relationship`, que utiliza o identificador único de cada um.

Essas relações podem ser vistas na modelagem lógica da Figura 28.

Figura 28 – Modelagem lógica das relações do banco de dados antes do preparador.



Fonte: elaborado pelo autor.

O ADES utiliza a variável da coluna `owner` para identificar quem tem permissão de acessar o lote de documentos, uma vez que a pessoa possui o link. Caso haja um dono identificado por essa coluna, somente ele poderá visualizar e manipular os documentos. Já no caso da coluna vazia, ou seja, com valor nulo, qualquer pessoa com o link consegue manipular o documento.

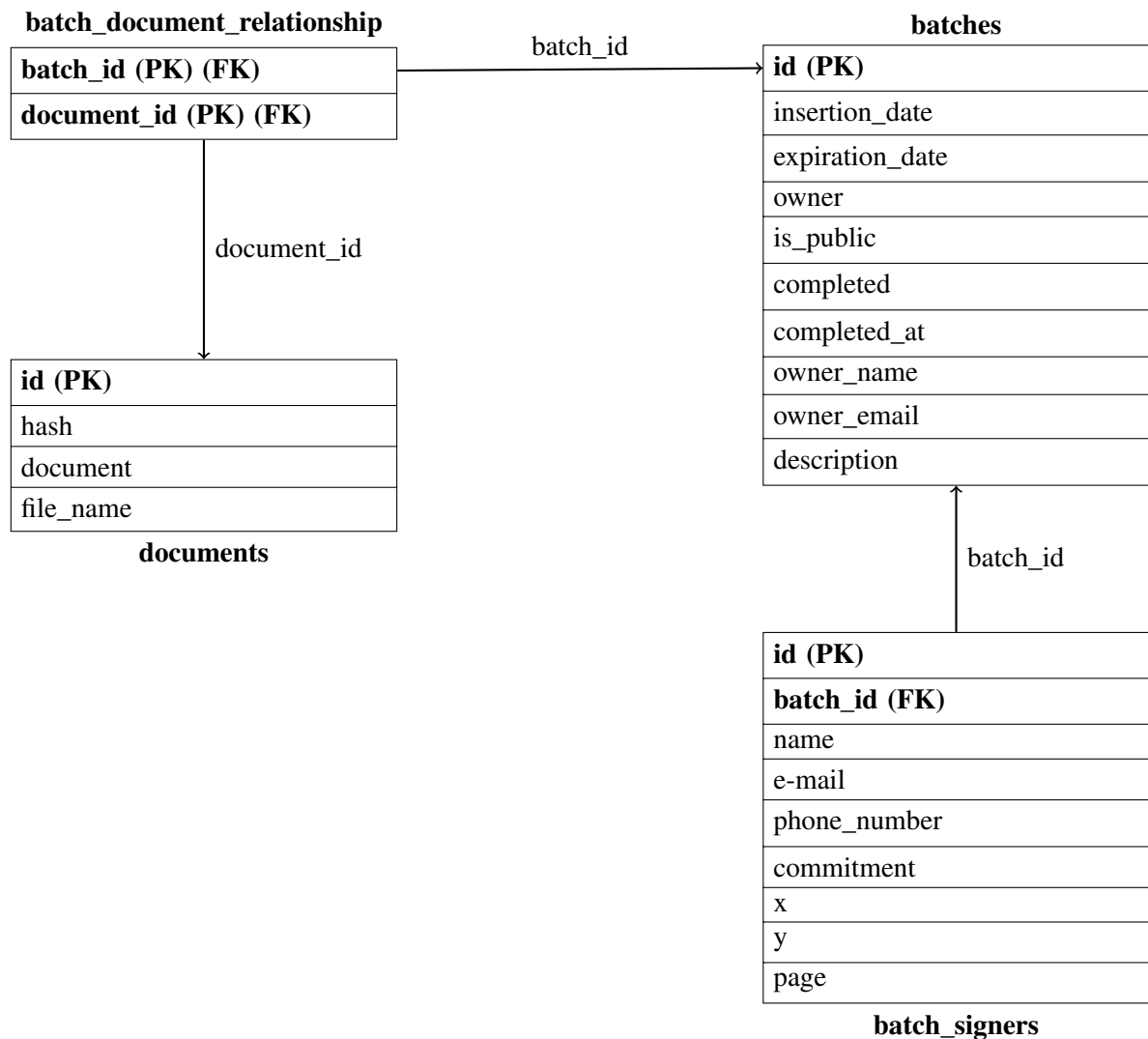
Para o preparador, é necessária uma adaptação para permitir lidar com todos requisitos funcionais do sistema. Assim, propõe-se as seguintes modificações no banco de dados, que podem ser vistas na modelagem lógica da Figura 29.

A tabela `batches` (lotes) foi ampliada com campos adicionais para representar, de maneira mais completa, o estado e o contexto de um documento preparado. Além do indicador de compartilhamento por link (`is_public`), o lote passa a registrar se o processo foi concluído (`completed`) e o instante em que a conclusão ocorreu (`completed_at`). Foram incluídos, ademais, campos descritivos do responsável pelo lote, como nome (`owner_name`) e e-mail (`owner_email`), além de um campo de descrição (`description`) para informações adicionais sobre o lote.

Adicionalmente, foi criada a tabela `batch_signers` para registrar quais são os signatários associados a cada lote e para permitir o monitoramento da progressão das assinaturas esperadas. O lote em que é signatário esperado é identificado pelo identificador do lote (`batch_id`). Para cada signatário, são armazenados dados de identificação, como nome, e-mail e/ou número de telefone (`phone_number`), e uma informação textual sobre seu compromisso no

processo (commitment). A tabela também contempla campos destinados à inserção do artefato visual de assinatura no PDF, registrando as coordenadas (x e y) e a página (page) onde a representação da assinatura deve ser posicionada. Por fim, o campo `signed_at` permite distinguir assinaturas pendentes (valor nulo) de assinaturas já coletadas (valor preenchido), registrando o momento em que o signatário efetivamente concluiu sua assinatura.

Figura 29 – Modelagem lógica das relações do banco de dados proposta para o preparador.



Fonte: elaborado pelo autor.

6.2.2 Criação de novas funções

Para atender aos requisitos funcionais do PdD, foram criadas diversas funções no GED que não existiam anteriormente no sistema e alteradas as existentes para atender aos novos requisitos. Essas funções podem ser agrupadas em diferentes categorias conforme suas responsabilidades.

Foram implementadas funções para gerenciar automaticamente o ciclo de vida dos documentos preparados. A função `complete_expired_public_prepared_batches` identifica

documentos preparados com compartilhamento por link que atingiram a data limite de expiração e os marca como completos. A função `complete_expired_private_prepared_batch` processa documentos privados que expiraram sem receber todas as assinaturas esperadas, enviando notificações aos responsáveis sobre a situação de incompletude. Por fim, a função `handle_prepared_batches_lifecycle` remove automaticamente o conteúdo binário dos PDFs de documentos preparados que já expiraram há mais de um período configurável, preservando apenas os metadados para fins de histórico e economizando espaço de armazenamento.

Para viabilizar as notificações, são necessárias funções que automatizem o envio de mensagens em diferentes momentos do processo. A função `notify_expected_signers_about_new_document` envia notificações aos signatários especificados quando um novo documento é preparado e está aguardando suas assinaturas, incluindo informações como prazo limite, descrição do documento e links para visualização e assinatura, por e-mail ou por WhatsApp conforme o meio de contato cadastrado. Cada link de convite embute um token único gerado para o signatário e o lote; ao ser acessado, o microserviço ID valida o token e autentica automaticamente o usuário, identificando-o para o restante do fluxo de assinatura. A função `handle_completion_notification` notifica o responsável pelo documento quando todas as assinaturas esperadas foram coletadas no fluxo com signatários especificados. A função `handle_incomplete_signatures_notification` envia alerta ao responsável quando um documento privado expirou sem receber todas as assinaturas esperadas, listando os signatários pendentes. Por fim, a função `notify_owner_signed_successfully` envia uma notificação especial quando o documento foi preparado e assinado apenas pelo próprio responsável, sem outros signatários esperados.

Para recuperar informações detalhadas sobre documentos preparados, a função `api_document_info` retorna informações completas de um documento preparado, incluindo metadados como datas de inserção e expiração, descrição, *status* de conclusão e lista de signatários com seus respectivos estados de assinatura. A função `list_user_documents` permite listar os documentos preparados por um usuário com suporte a paginação, ordenação por data ou nome, e busca por termo, retornando informações resumidas de cada documento para exibição em lista.

Quanto à verificação de permissões de acesso e de se usuários são signatários esperados, a função `is_user_expected_signer` verifica se um usuário autenticado corresponde a um dos signatários cadastrados para o lote, comparando o e-mail fornecido pelo IdP com o registrado em `batch\_signers`. Essa verificação orienta tanto o controle de acesso no fluxo restrito quanto o posicionamento do artefato visual quando alguém assina a partir de um link compartilhado: signatários esperados autenticados têm sua assinatura ancorada nas coordenadas definidas no preparo e o campo `signed\_at` é atualizado; demais usuários autenticados com o link podem assinar, mas sem reutilizar posições reservadas. A função `has_access`, que já existia no sistema, foi estendida para considerar se o lote permite acesso via link (`is\_public`), se o usuário é o proprietário (`owner`) ou se é signatário esperado. Com `is\_public` ativado, qualquer usuário autenticado que possua o link obtém acesso; com `is\_publ`

ic desativado, apenas o proprietário e os signatários esperados podem visualizar e assinar o documento preparado.

Para processar documentos no fluxo do preparador, a função `create_batch_from_uploaded_pdfs` foi adaptada para processar uploads de documentos preparados, incluindo a criação de batches com metadados adicionais como descrição, informações do responsável e configuração de signatários esperados. A função `insert_signers` insere os signatários especificados na tabela `batch_signers`, incluindo nome, e-mail e/ou número de telefone, comprometimento e posições de assinatura no PDF. A função `register_user_signature_for_batch` registra quando um signatário realiza sua assinatura, atualizando o campo `signed_at` e verificando se todas as assinaturas esperadas foram coletadas para marcar o documento como completo. A função `handle_batch_completion` processa a conclusão de um documento quando todas as assinaturas esperadas foram recebidas, atualizando o *status* e disparando notificações apropriadas.

Para verificar estados e condições dos documentos preparados, a função `is_batch_completed` verifica se um documento preparado foi concluído. A função `are_there_expected_signatures_remaining` verifica se ainda há assinaturas pendentes para um documento, comparando signatários esperados com aqueles que já assinaram. A função `is_within_deadline_for_signatures` verifica se a data atual ainda está dentro do prazo limite configurado para recebimento de assinaturas, considerando a data limite especificada.

7 RESULTADOS

Este capítulo apresenta os resultados da implementação do PdD por meio do registro de telas criadas para a aplicação e o contexto em que aparecem e são utilizadas. Aqui demonstra-se que a implementação realizada os materializa, com capturas da interface em um cenário de uso concreto. A verificação sistemática do atendimento aos requisitos funcionais, com rastreabilidade entre o requisito e o registro de que é atendido é apresentada no Capítulo 8.

7.1 DEMONSTRAÇÃO DO FLUXO PRINCIPAL

Para exemplificar um caso que cobre a maior parte das funcionalidades, utiliza-se um documento com duas assinaturas esperadas: uma de quem prepara e outra de um signatário adicional. Quem prepara aguarda a coleta da segunda assinatura e a notificação de conclusão para obter o documento finalizado; o signatário esperado assina após receber a notificação por e-mail. As etapas a seguir correspondem ao fluxo principal proposto na Figura 17 e detalhado nas Figuras 18 a 24.

7.1.1 Selecionar novo documento

Na etapa de seleção do documento (Figura 19), o fluxo parte da página principal do PAS (Figura 31) com um clique no botão de *Assinar*. Caso não esteja autenticado, o sistema solicita *login* (Figura 3). Em seguida, é exibida a lista de documentos já preparados pelo usuário (Figura 32), inicialmente vazia. Ao clicar no botão de *Novo documento*, o usuário escolhe um documento do tipo PDF do seu dispositivo (Figura 33) e o sistema apresenta a tela de preparo, onde poderá especificar signatários e demais opções para o documento (Figuras 34).

O documento utilizado no exemplo espera duas assinaturas, sendo uma delas do próprio usuário que está preparando o documento. O PDF de exemplo utilizado ao longo desse fluxo está ilustrado na Figura 30.

Figura 30 – Documento PDF de exemplo utilizado nos testes do fluxo.

Contrato

CONTRATANTE: Victor Camargo

CONTRATADA: João da Silva

E por estarem de acordo, as partes assinam o presente contrato em duas vias de igual teor e forma, na presença de duas testemunhas.

Florianópolis, 1 de Julho de 2025

Victor Camargo

João da Silva

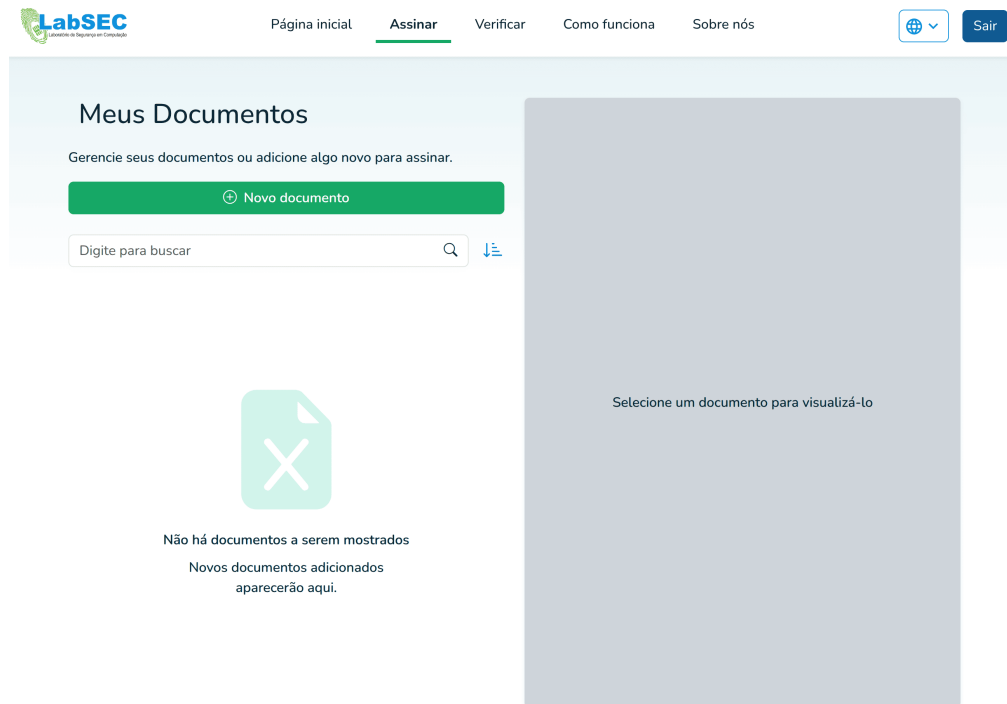
Fonte: elaborado pelo autor.

Figura 31 – Nova tela inicial do Portal de Assinatura.



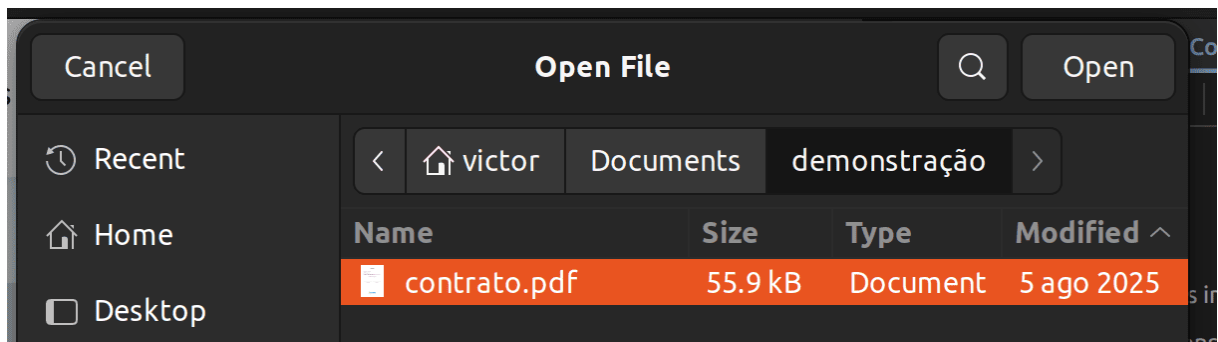
Fonte: elaborado pelo autor.

Figura 32 – Lista de documentos do usuário vazia e com botão para novo documento.



Fonte: elaborado pelo autor.

Figura 33 – Seleção do arquivo PDF para novo documento no sistema operacional Ubuntu.



Fonte: elaborado pelo autor.

Figura 34 – Tela de preparo do documento: estado inicial após selecionar documento novo, com campos e botões para *Assinar* ou *Compartilhar*, usados nos fluxos das seções seguintes.

Fonte: elaborado pelo autor.

7.1.2 Adicionar signatários ao documento

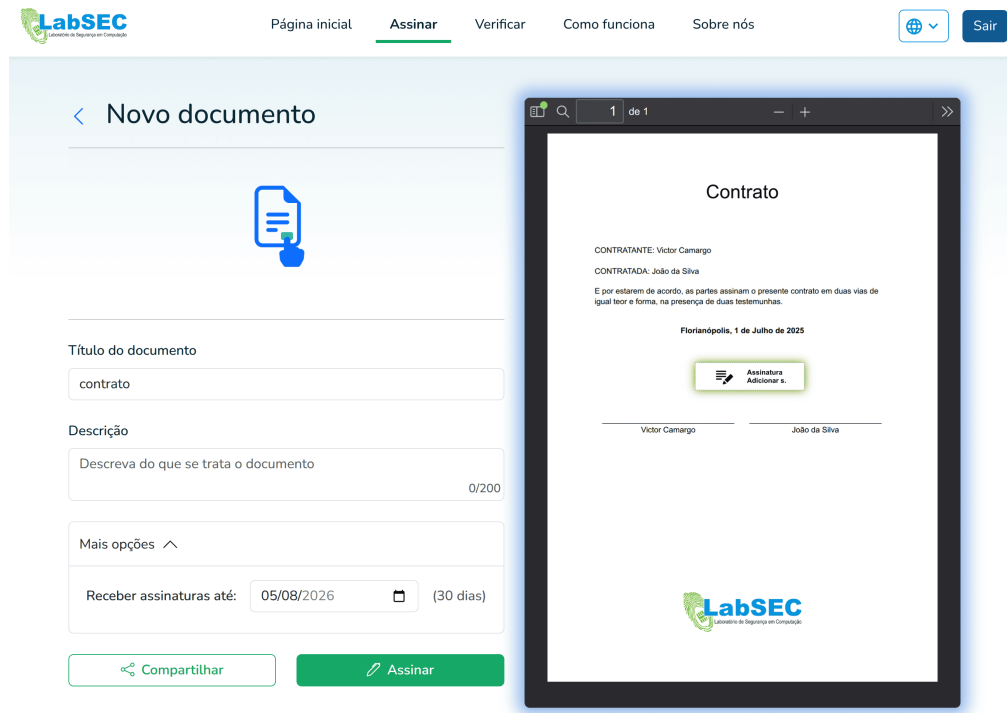
Na etapa de inclusão de signatários (Figura 20), a interface indica que o usuário pode clicar sobre o PDF para posicionar assinaturas; em dispositivos com ponteiro, um retângulo acompanha o cursor para mostrar onde o artefato visual será colocado (Figura 35).

Ao clicar no local desejado onde ficará a assinatura no documento, abre-se a janela para inserir dados do signatário (Figura 36). Nele, preenchem-se nome, meio de contato do signatário (e-mail ou número de telefone para WhatsApp), comprometimento, e confirma-se com *Salvar*. O botão *Sou eu* preenche automaticamente nome e e-mail do usuário autenticado, o que facilita o processo quando quem está preparando o documento é também o signatário esperado (Figura 37). Ao salvar, pode ser visto o signatário listado à esquerda do documento e a posição em que ficará seu artefato visual no documento (Figura 38). Continuando o preparo do documento exemplo, o processo de adicionar signatário pode ser repetido, adicionando um segundo signatário ao documento (Figura 39).

Ainda nessa tela, o usuário pode ajustar nome de exibição do documento, descrição e data limite para recebimento das assinaturas (Figura 39), ou manter os valores padrão.

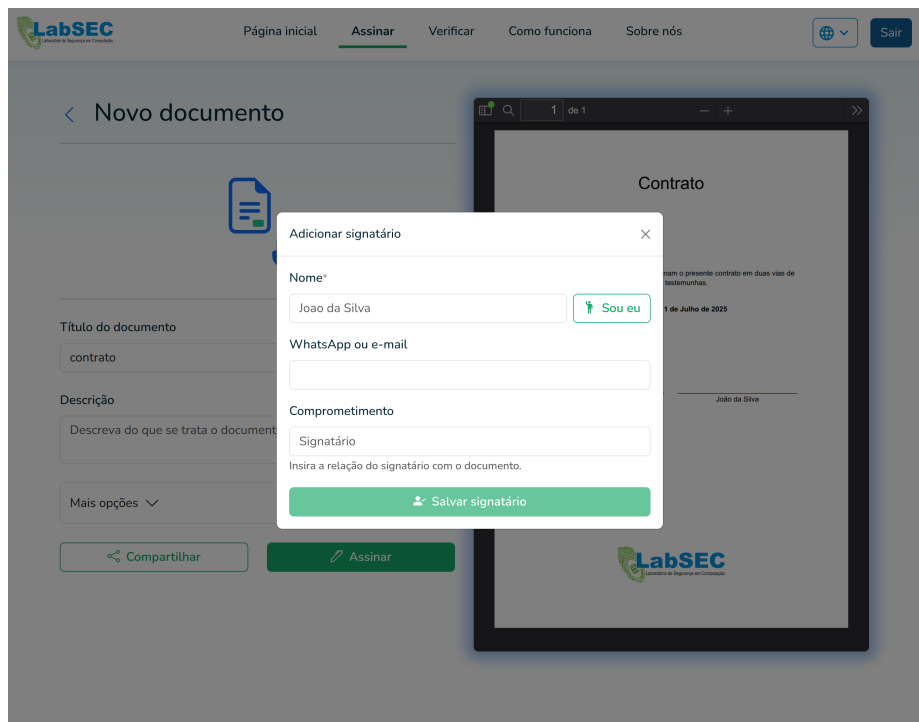
No exemplo, há dois signatários esperados, sendo um deles o próprio usuário que está preparando.

Figura 35 – Indicação visual para posicionar o artefato de assinatura sobre o PDF.



Fonte: elaborado pelo autor.

Figura 36 – Formulário de dados do signatário no preparo do documento.



Fonte: elaborado pelo autor.

Figura 37 – Formulário de dados do signatário no preparo do documento após clicar em *Sou eu*.

The screenshot shows the LabSEC web application interface. The main page is titled 'Novo documento' and includes a sidebar with options like 'Compartilhar' and 'Assinar'. A modal window titled 'Adicionar signatário' is open, displaying the following fields:

- Nome***: Victor Camargo (Status: Preenchido)
- WhatsApp ou e-mail**: victor@email.com
- Comprometimento**: Signatário (Instrução: Insira a relação do signatário com o documento.)

A green button labeled 'Salvar signatário' is at the bottom of the modal. In the background, a document titled 'Contrato' is visible, showing a signature line for 'João da Silva'.

Fonte: elaborado pelo autor.

Figura 38 – Tela de preparo após o usuário se adicionar como signatário.

The screenshot shows the LabSEC web application interface after a user has been added as a signatory. The main page is titled 'Novo documento' and includes a sidebar with options like 'Assinar'. The document preparation screen shows the following details:

- 1 assinatura posicionada**: Victor Camargo (você) with an 'Assinar' button and a checkmark.
- Título do documento**: contrato
- Descrição**: Descreva do que se trata o documento (0/200)
- Mais opções**: Receber assinaturas até: 05/08/2026 (30 dias)

A green button labeled 'Assinar' is at the bottom. In the background, a document titled 'Contrato' is visible, showing the signature of 'Victor Camargo' and the signature line for 'João da Silva'.

Fonte: elaborado pelo autor.

Figura 39 – Documento com 2 signatários adicionados, com marcadores de assinatura sobre o documento e nome e descrição definidos.

The screenshot displays the 'Novo documento' (New document) interface of the LabSEC system. The interface is divided into two main sections: a form on the left and a document preview on the right.

Form Section (Left):

- 2 assinaturas posicionadas:** Two signatories are listed: 'Victor Camargo (você)' with an 'Assinar' (Sign) button and a checkmark, and 'João da Silva'.
- Limpar assinaturas:** A link to clear the signatures.
- Título do documento:** A text field containing 'contrato'.
- Descrição:** A text field containing 'Contrato exemplo' with a character count of '16/200'.
- Mais opções:** A dropdown menu.
- Receber assinaturas até:** A date field set to '05/08/2026' with a calendar icon and a duration of '(30 dias)'.
- Enviar:** A large green button at the bottom.

Document Preview Section (Right):

- The preview shows a document titled 'Contrato'.
- It lists the 'CONTRATANTE: Victor Camargo' and 'CONTRATADA: João da Silva'.
- A note states: 'E por estarem de acordo, as partes assinam o presente contrato em duas vias de igual teor e forma, na presença de duas testemunhas.'
- The date is 'Florianópolis, 1 de Julho de 2025'.
- Two signature blocks are shown: 'Assinatura Victor C.' and 'Assinatura João S.', each with a signature icon and the name below it.
- The LabSEC logo is at the bottom.

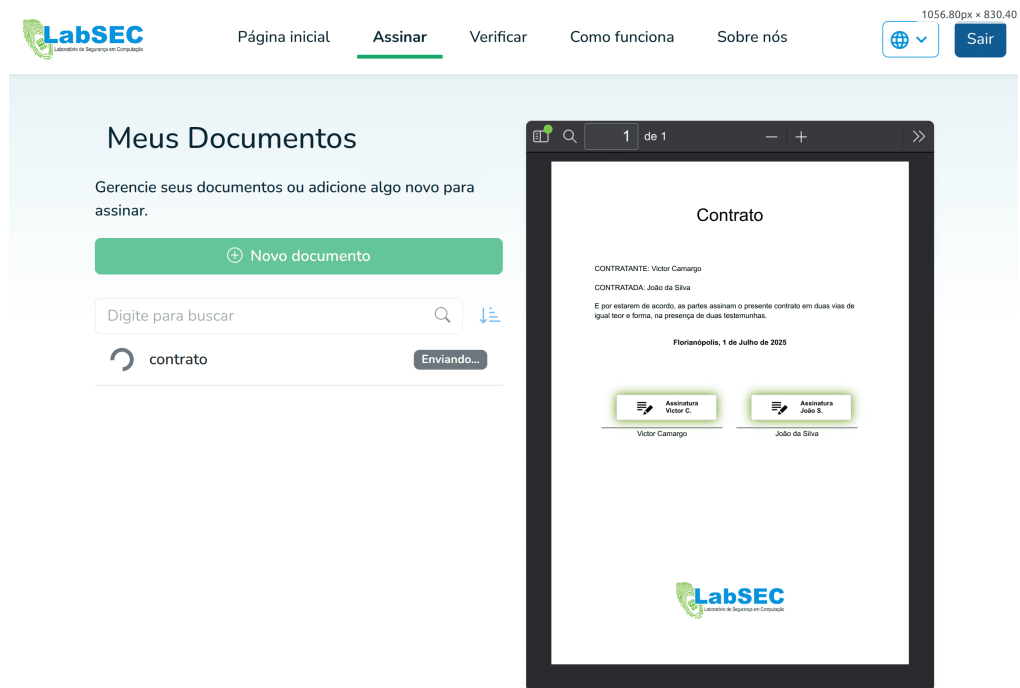
Fonte: elaborado pelo autor.

7.1.3 Envio do documento e finalização do preparo

Com tudo definido, o usuário clica em *Enviar*. O sistema processa o envio (Figura 40), retornando o usuário à lista de documento, e exibe confirmação (Figura 41) ao completar a operação de envio. Nesse passo, o sistema já realiza a assinatura do usuário que preparou, desde que ele constasse entre os signatários esperados do documento.

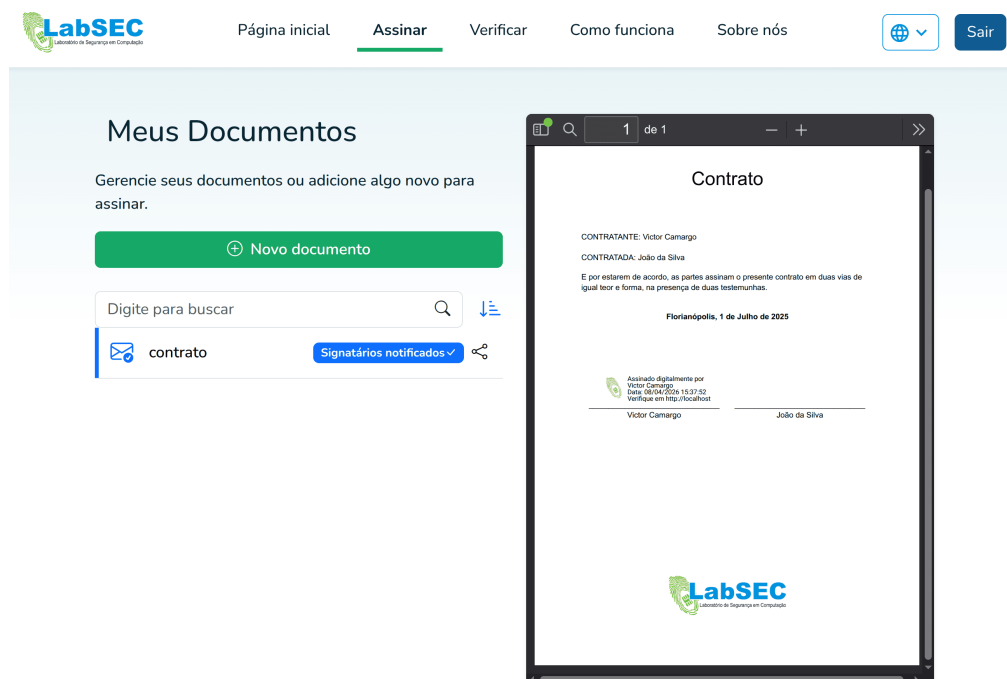
Ao clicar no item recém preparado na lista de documentos, a interface abrirá uma tela que mostra o andamento da coleta de assinaturas, prazo e quem já assinou (Figura 42). Voltando à lista, visualizam-se todos os documentos preparados e o estado de cada um (Figura 43).

Figura 40 – Estado de processamento após o envio do documento.



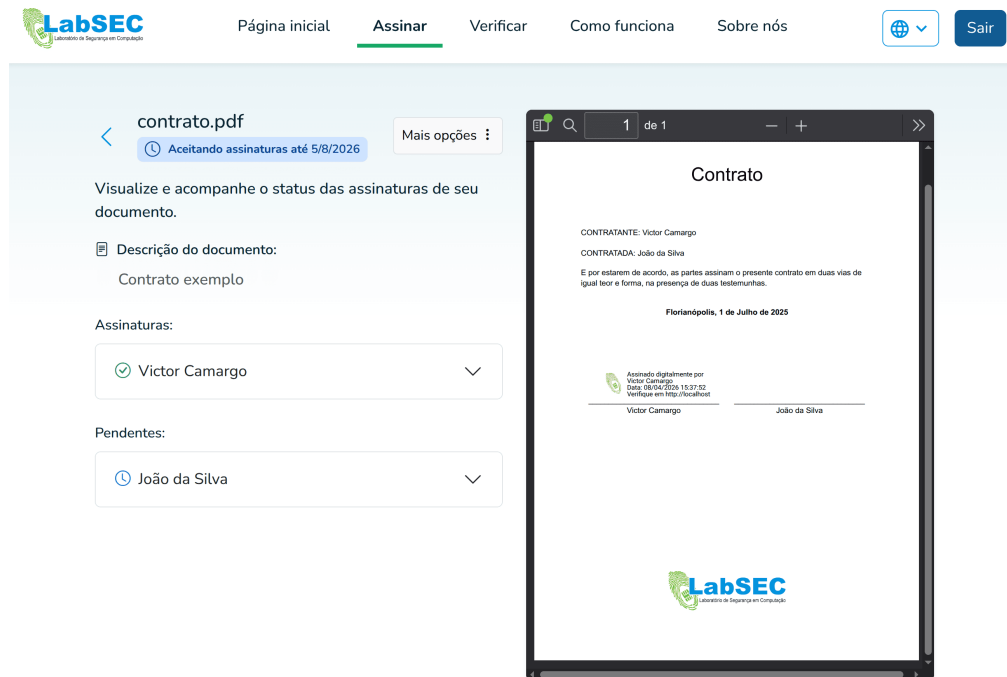
Fonte: elaborado pelo autor.

Figura 41 – Confirmação de sucesso após o preparo do documento.



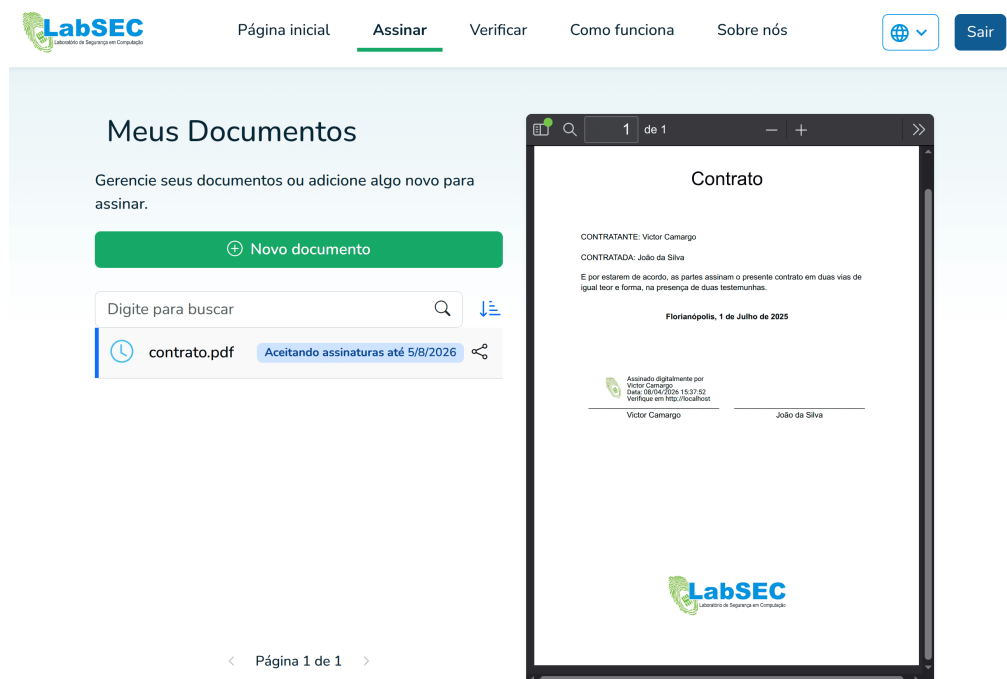
Fonte: elaborado pelo autor.

Figura 42 – Monitoramento do progresso de coleta de assinaturas de um documento preparado.



Fonte: elaborado pelo autor.

Figura 43 – Lista de documentos preparados com indicação do estado de completude.



Fonte: elaborado pelo autor.

7.1.4 Assinatura do signatário a partir do e-mail

Na etapa de coleta de assinaturas (Figura 21), a assinatura a partir do e-mail (Figura 22) inicia quando o signatário recebe mensagem com dados do documento (nome, descrição, quem preparou, comprometimento) e botões para assinar de imediato ou visualizar o documento no *site* primeiramente (Figura 44). Vindo do botão de assinar, o sistema conduz a assinatura (Figura 45) até a conclusão (Figura 46). Optando por visualizar primeiro, o documento e os signatários são exibidos (Figura 47), podendo a assinatura ser feita em seguida.

O link presente na notificação contém um token único associado ao signatário convidado. Ao clicar no botão, o navegador aciona o microsserviço ID, que valida o token e autentica automaticamente o usuário, sem exibir a tela de login no IdP. O sistema reconhece, assim, qual signatário esperado está assinando e aplica o artefato visual na posição definida no preparo. Essa autenticação baseia-se na premissa de que quem tem acesso ao e-mail ou à mensagem recebida no celular detém um canal pessoal de comunicação, tratado como fator de posse sobre dados pessoais do destinatário.

Figura 44 – Email de notificação para signatário esperado.



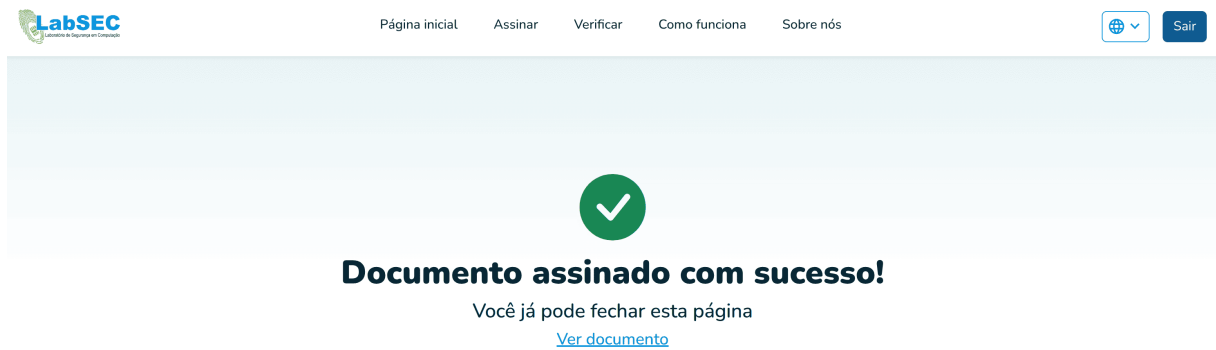
Fonte: elaborado pelo autor.

Figura 45 – Tela do processo de assinatura para o signatário.



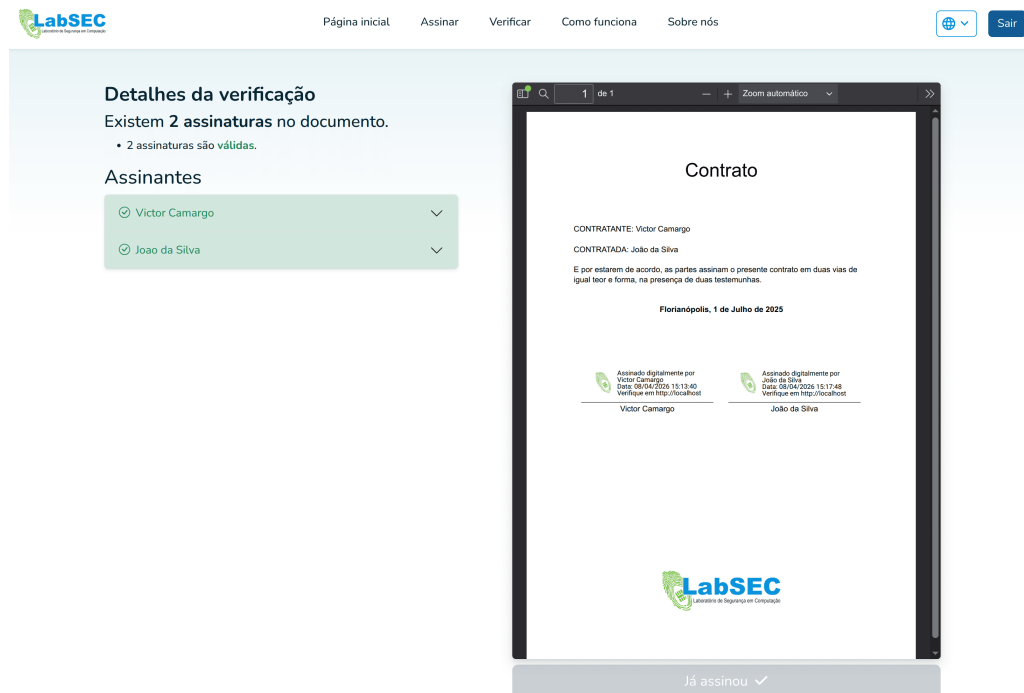
Fonte: elaborado pelo autor.

Figura 46 – Tela após a conclusão da assinatura pelo signatário esperado.



Fonte: elaborado pelo autor.

Figura 47 – Visualização do documento e lista de signatários antes ou após assinar.



Fonte: elaborado pelo autor.

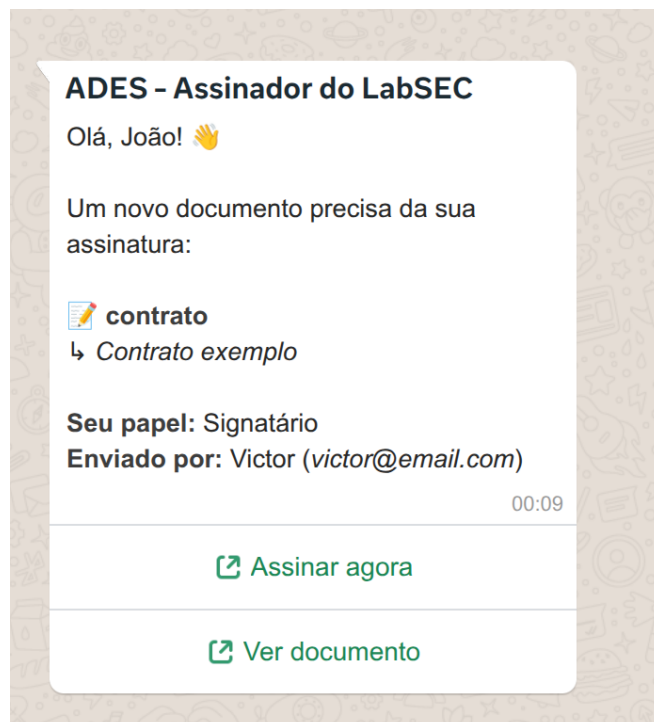
7.1.5 Notificação de pedido de assinatura por WhatsApp

Quando o cadastro informa o número de telefone do signatário, a notificação de pedido de assinatura é enviada por WhatsApp (Figura 48). Após o envio do documento, o signatário recebe mensagem no aplicativo com dados do pedido de assinatura e link para acessar o documento (Figura 49), com o mesmo mecanismo de token único e autenticação automática via microserviço ID descrito para as notificações por e-mail.

Figura 48 – Formulário de cadastro de signatário com meio de contato (e-mail ou número de telefone para WhatsApp).

Fonte: elaborado pelo autor.

Figura 49 – Mensagem de pedido de assinatura recebida pelo signatário no WhatsApp.

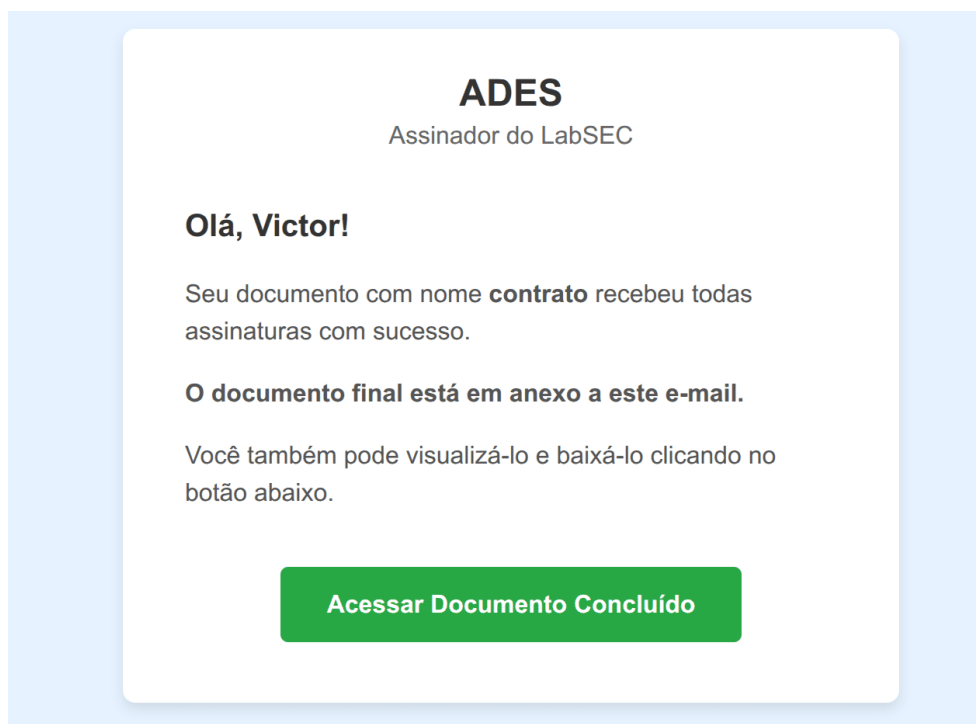


Fonte: elaborado pelo autor.

7.1.6 Recebimento do documento finalizado

Na etapa de recebimento do documento finalizado (Figuras 23 e 24), quem preparou o documento recebe uma notificação por e-mail (Figura 50) contendo o documento finalizado em anexo, e link para ele no *site*. Neste momento, o usuário já pode baixar o anexo, assim completando o processo.

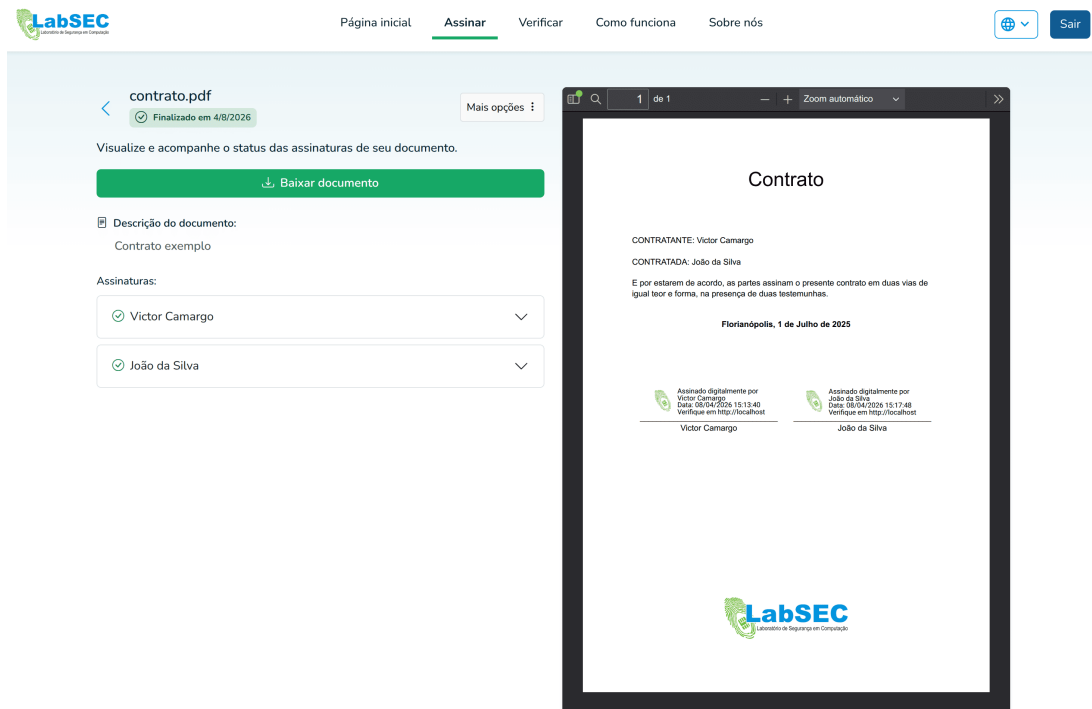
Figura 50 – Email de notificação para quem preparou o documento.



Fonte: elaborado pelo autor.

Alternativamente, ele pode querer visualizá-lo no sistema primeiro, clicando no link para visualizá-lo, autenticando-se e podendo baixá-lo nessa mesma página clicando o botão de "Baixar"(Figura 51).

Figura 51 – Tela de visualização do documento finalizado.



Fonte: elaborado pelo autor.

7.2 DEMONSTRAÇÃO DE COMPORTAMENTOS ALTERNATIVOS

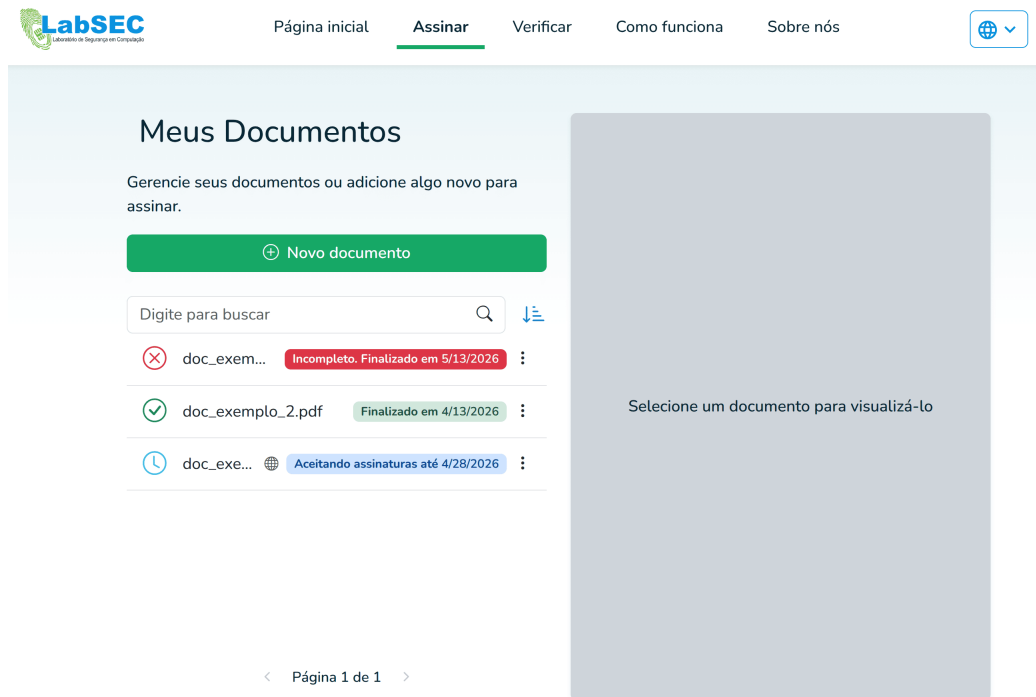
As seções a seguir apresentam evidências dos fluxos alternativos propostos na Seção 5.2.

7.2.1 Monitoramento do estado do documento

Na lista de documentos, observa-se a distinção entre estados previstos na proposta (concluído, pendente e incompleto).

Caso tenha recebido as assinaturas esperadas, é mostrado como concluído. Caso esteja aguardando assinaturas, é mostrado como pendente. Caso não tenha recebido todas as assinaturas esperadas, é mostrado como incompleto. A Figura 52 mostra uma lista com documentos nesses estados.

Figura 52 – Lista de documentos com documentos em diversos estados de completude.



Fonte: elaborado pelo autor.

7.2.2 Documentos que não receberam todas assinaturas esperadas

Caso um documento preparado pelo usuário não tenha recebido todas assinaturas esperadas dentro do prazo previsto, o usuário receberá uma notificação por e-mail avisando-o sobre isso (Figura 53), sem o arquivo em anexo, para evitar que o usuário baixe por acidente o documento acreditando que ele foi concluído com sucesso.

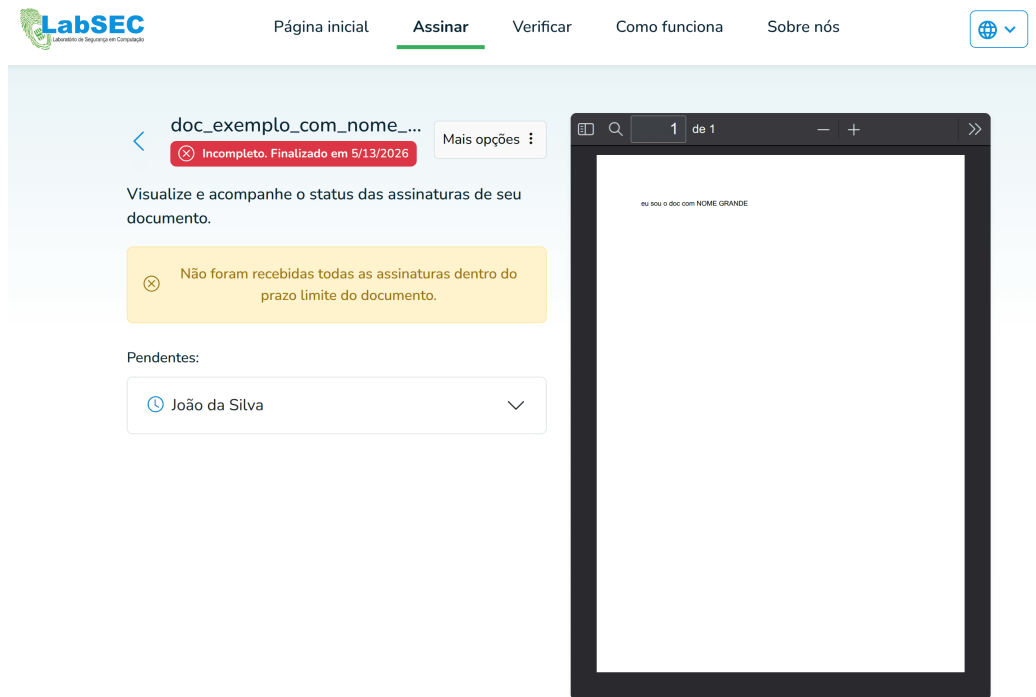
Figura 53 – Email de notificação para quem preparou o documento.



Fonte: elaborado pelo autor.

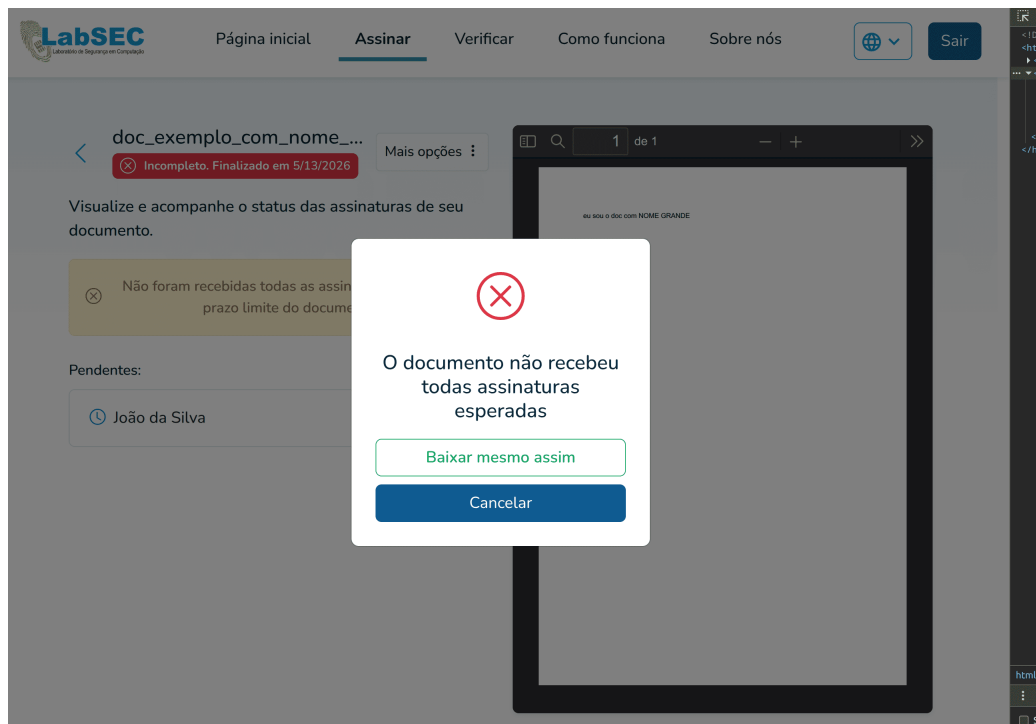
Porém, o usuário ainda poderá baixar o documento incompleto acessando a página do documento (Figura 54), que destaca esse estado de inconclusão. Para isso, precisa abrir o menu de "Mais opções" e clicar em "Baixar", que o apresentará primeiramente um alerta de que o documento não foi concluído com sucesso e pedirá sua confirmação (Figura 55), para garantir que esteja ciente sobre a situação.

Figura 54 – Tela de visualização do documento finalizado incompleto.



Fonte: elaborado pelo autor.

Figura 55 – Aviso prévio sobre situação atual do documento incompleto antes de baixar.

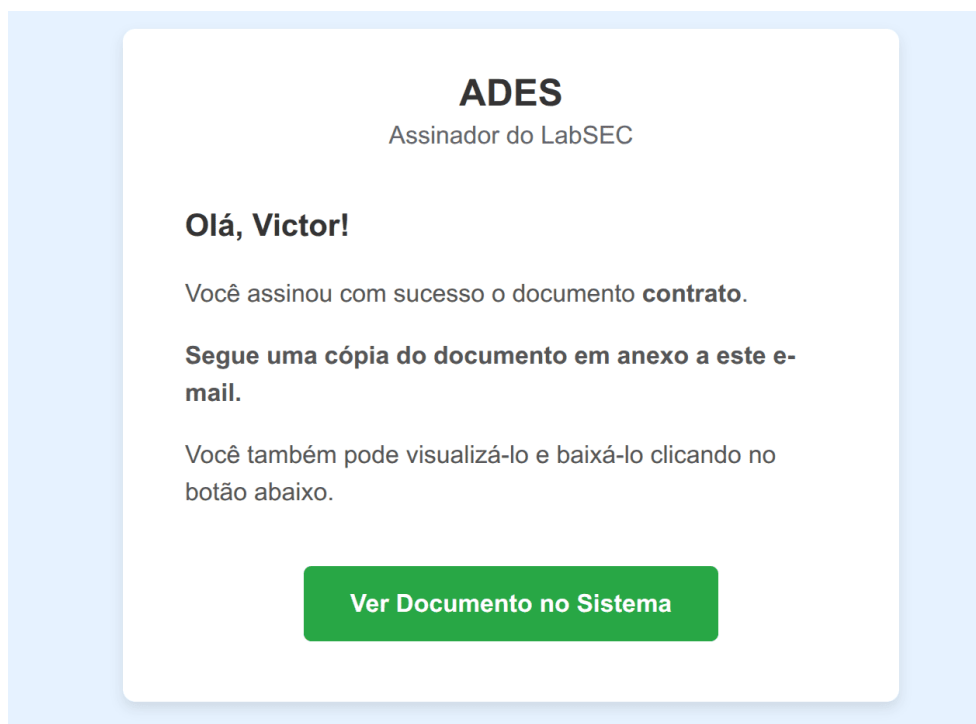


Fonte: elaborado pelo autor.

7.2.3 Assinatura com 1 só clique sem artefato visual

A evidência corresponde ao fluxo da Figura 25: o usuário seleciona o documento e aciona *Assinar* na tela de preparo inicial (Figura 34), sem posicionar artefato visual. Quem prepara e é o único signatário recebe e-mail de confirmação (Figura 56).

Figura 56 – Email de confirmação de assinatura concluída para o signatário.

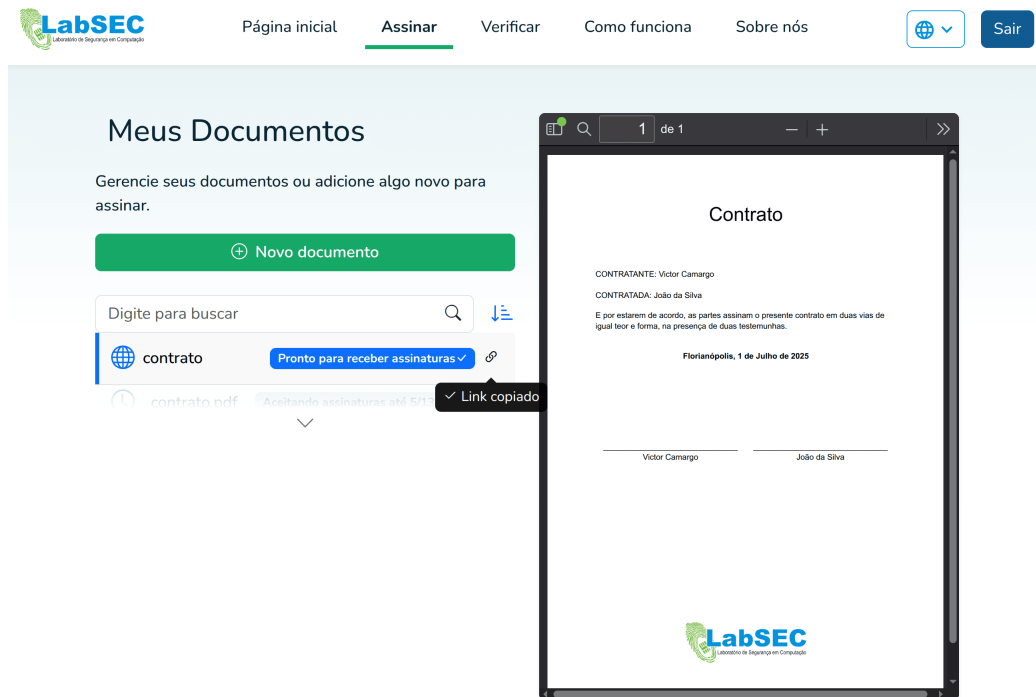


Fonte: elaborado pelo autor.

7.2.4 Compartilhar documento por link

A evidência corresponde ao fluxo da Figura 26: após selecionar o documento, o usuário aciona *Compartilhar* na página de preparo (Figura 34) e obtém o link para encaminhar aos signatários (Figura 57). O mesmo mecanismo pode ser usado após cadastrar signatários esperados e posicionar artefatos visuais: o link gerado permite que qualquer pessoa autenticada assine, enquanto o sistema identifica, pelo e-mail de login, se quem acessa era um dos signatários nomeados no preparo, aplicando nesse caso a posição predefinida da assinatura.

Figura 57 – Link de compartilhamento gerado para assinatura por terceiros.

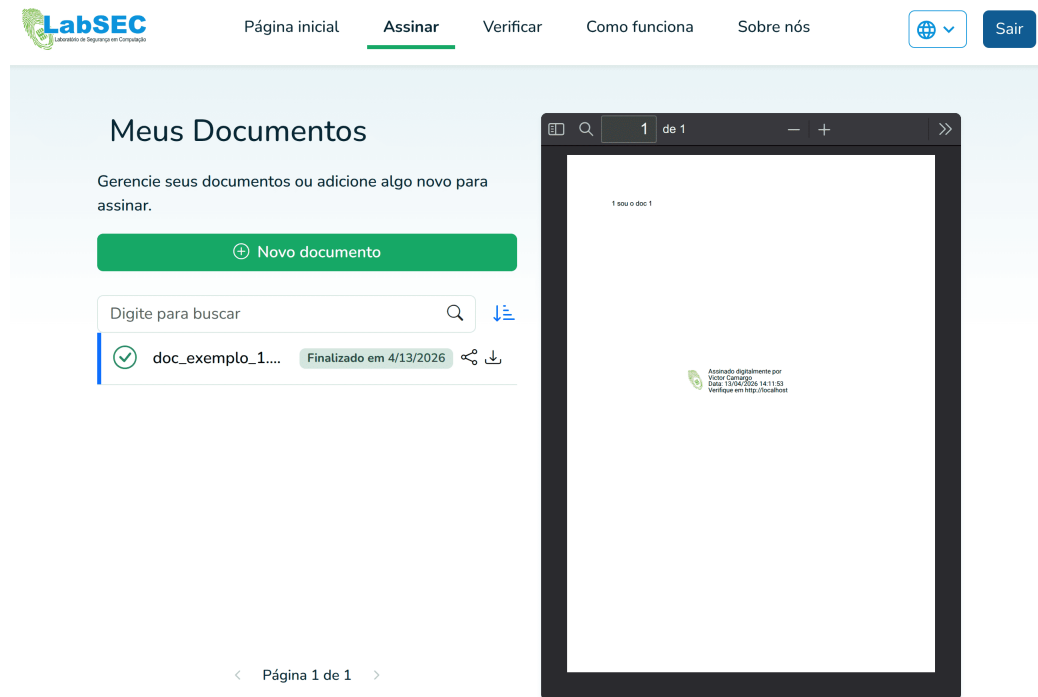


Fonte: elaborado pelo autor.

7.2.5 Compartilhar documentos já inseridos no sistema

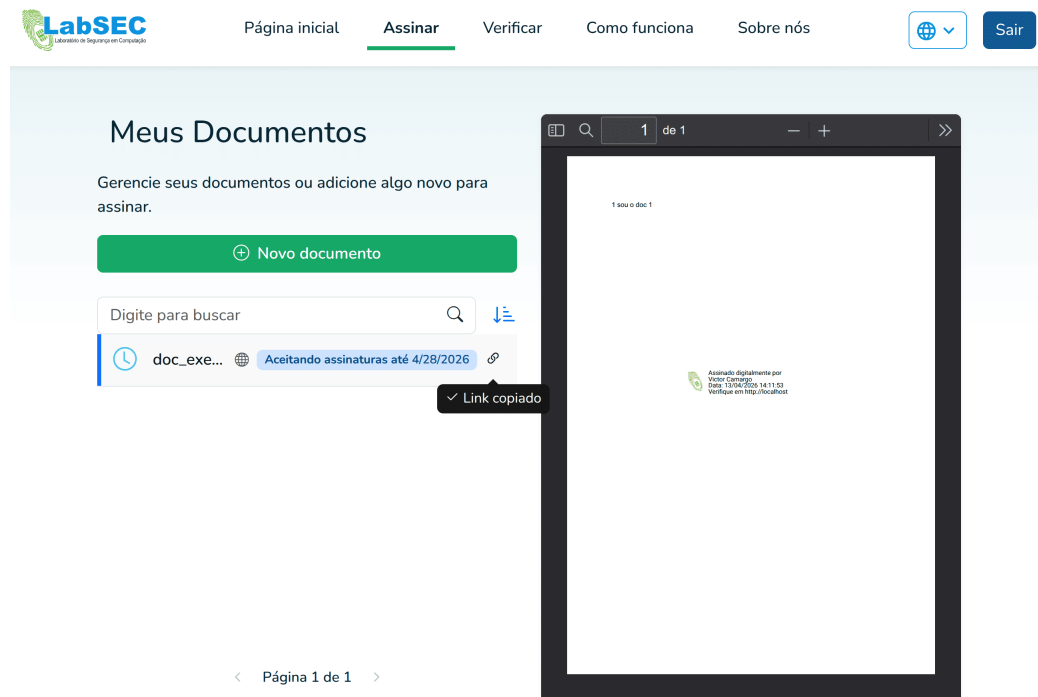
A evidência corresponde ao fluxo da Figura 27: na lista de documentos, o usuário seleciona um documento já preparado (Figura 58) e aciona o compartilhamento por link (Figura 59).

Figura 58 – Documento na lista de documentos já finalizado e que pode ser compartilhado por link.



Fonte: elaborado pelo autor.

Figura 59 – Compartilhamento por link de documento já inserido no sistema.



Fonte: elaborado pelo autor.

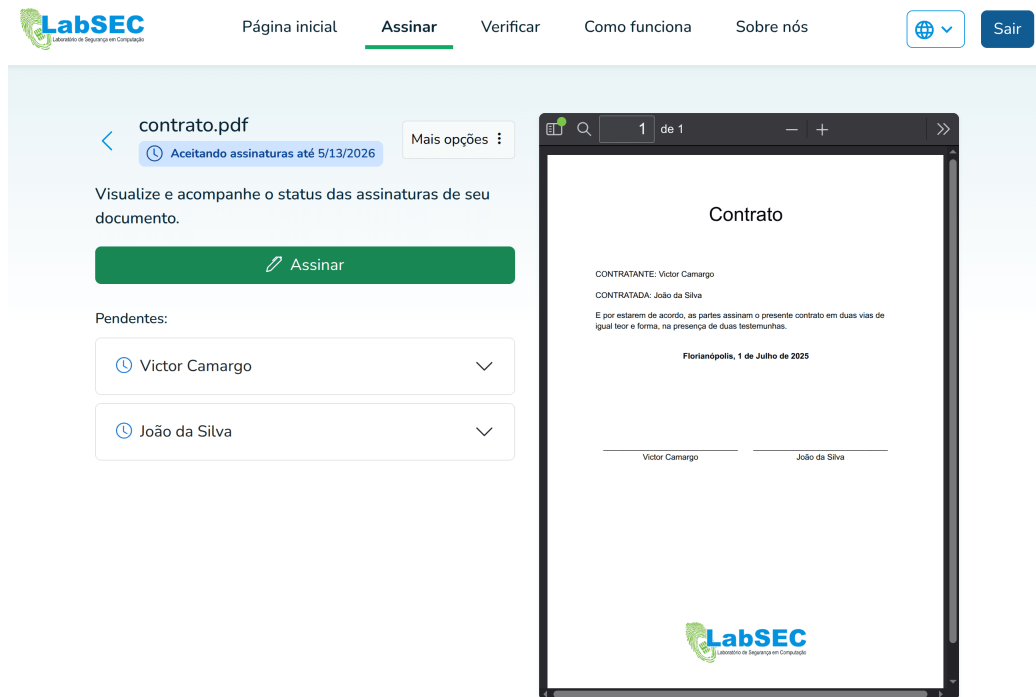
7.2.6 Adiar a própria assinatura em um documento preparado em que o usuário é um signatário especificado

A evidência ilustra a opção prevista na proposta de adiar a própria assinatura: o usuário desmarca a caixa *Assinar* ao lado do seu nome durante o preparo (Figura 60). Após enviar o documento, poderá assinar a qualquer momento na página de detalhes do documento, onde encontrará um botão de "Assinar"(Figura 61) e ao clicá-lo realizará sua assinatura (Figura 62).

Figura 60 – Opção de desmarcar a assinatura do próprio usuário ao concluir o preparo.

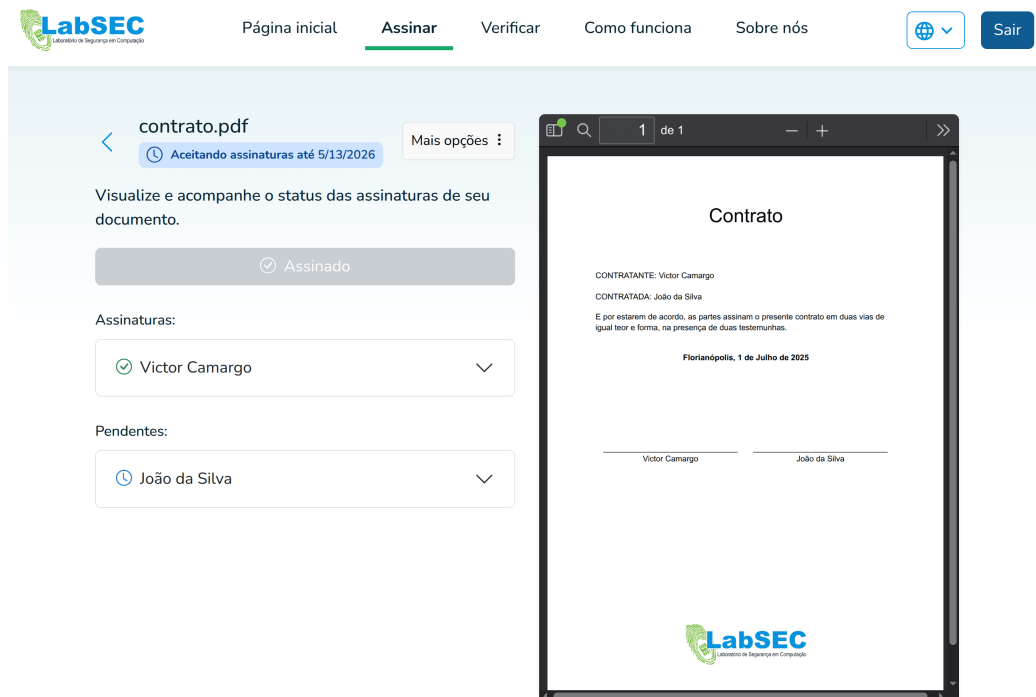
Fonte: elaborado pelo autor.

Figura 61 – Assinatura posterior do documento pelo usuário que preparou.



Fonte: elaborado pelo autor.

Figura 62 – Assinatura posterior do documento pelo usuário que preparou.



Fonte: elaborado pelo autor.

8 ATENDIMENTO AOS REQUISITOS FUNCIONAIS

Este capítulo consolida a verificação de que a implementação do PdD atende aos requisitos funcionais definidos no Capítulo 4. Com base nas evidências apresentadas no Capítulo 7, a Tabela 4 relaciona cada requisito ao comportamento observado na aplicação e às figuras que o demonstram. Os parágrafos seguintes sintetizam o atendimento por grupos de funcionalidades.

Com as implementações realizadas tanto no PAS quanto no GED, observa-se que o PdD atende ao conjunto de requisitos funcionais especificados.

Tabela 4 – Atendimento aos Requisitos Funcionais do Preparador de Documentos.

Requisito (ID)	Como é atendido	Evidências
RF1	Cadastro de signatários com nome, e-mail e/ou número de telefone e campos de descrição/comprometimento durante o preparo do documento.	Figura 36; Figura 48
RF2	Possibilidade de o usuário preparar um documento incluindo a si mesmo como signatário.	Figura 39
RF3	Possibilidade de preparar documentos sem incluir a própria assinatura no fluxo.	Figura 39
RF4	Acompanhamento do <i>status</i> de coleta de assinaturas (assinaturas recebidas/pendentes e prazo).	Figura 42
RF5	Envio de notificação de assinatura requerida a signatários esperados por e-mail.	Figura 44
RF6	Envio de notificação de assinatura requerida por WhatsApp quando o cadastro informa número de telefone.	Figura 48; Figura 49
RF7	Notificação ao usuário que preparou quando todas as assinaturas esperadas são coletadas.	Figura 50
RF8	Configuração de data limite para recebimento de assinaturas durante o preparo.	Figura 39
RF9	Definição da posição do artefato visual de assinatura no PDF para signatários esperados.	Figura 35
RF10	Apresentação de estados intermediários do documento (recebendo assinaturas, incompleto, completo).	Figura 52

Requisito (ID)	Como é atendido	Evidências
RF11	Preparo e compartilhamento de documentos por link (sem signatários esperados; quem possui o link pode assinar).	Figura 34; Figura 57
RF12	Assinatura do signatário esperado com um clique após autenticação.	Figura 45
RF13	Definição de mensagem/descrição para contextualizar o signatário na notificação.	Figura 44
RF14	Quando o preparador é o único signatário e assina ao final do preparo, disponibilização de download com um clique.	Figura 41
RF15	Disponibilização de link para acesso ao documento ao final do preparo, para compartilhamento externo.	Figura 57
RF16	download do documento preparado a qualquer momento, com aviso quando o documento não está concluído.	Figura 55
RF17	Notificação ao usuário que preparou quando o prazo limite é atingido sem coletar todas as assinaturas esperadas.	Figura 53
RF18	Inserção de descrição e comprometimento de signatários no cadastro, e apresentação desses dados no fluxo de notificação/assinatura.	Figura 36; Figura 44
RF19	Assinatura do preparador com um só clique sem posicionamento de artefato visual após inserir documento.	Figura 25; Figura 34
RF20	Obtenção/compartilhamento por link de documento já preparado/inserido no sistema.	Figura 59
RF21	Adição do próprio usuário como signatário esperado sem assinar ao final do preparo, permitindo assinar posteriormente.	Figura 60; Figura 61

No fluxo de preparo, a etapa de especificação de signatários permite cadastrar participantes com e-mail ou número de telefone (RF1 e RF18), incluindo a definição do comprometimento, e também permite que o usuário se adicione como signatário (RF2). Além disso, o usuário pode optar por preparar o documento sem incluir sua própria assinatura (RF3) e, quando desejado, adiar sua assinatura mesmo estando entre os signatários esperados (RF21). A tela de preparo também suporta o posicionamento do artefato visual no PDF para signatários

esperados (RF9) e a configuração de prazo limite para recebimento de assinaturas (RF8). Ao final do preparo, o sistema disponibiliza um link para compartilhamento do documento (RF15) e, quando necessário, permite obter posteriormente um link para documentos já inseridos no sistema (RF20). O fluxo contempla, ainda, a preparação de documentos para assinatura apenas por compartilhamento de link, sem signatários esperados (RF11). Por fim, são tratados casos específicos, como o preparo de documento em que o próprio usuário é o único signatário esperado, com assinatura facilitada e botão para download ao final do preparo (RF14), e a assinatura com um só clique sem posicionamento de artefato visual (RF19).

Quanto às notificações, o canal depende do meio de contato cadastrado: com e-mail, o sistema envia a demanda de assinatura por esse canal (RF5); com número de telefone, por WhatsApp (RF6). Em ambos os casos, utiliza-se a descrição para contextualizar o usuário sobre o documento (RF13), bem como a apresentação de informações como comprometimento (relacionadas a RF1 e RF18), como ilustrado nas Figuras 44 e 49. Após autenticação, o signatário esperado consegue realizar a assinatura com apenas um clique (RF12). Para quem prepara o documento, há notificações quando todas as assinaturas esperadas são coletadas (RF7) e quando o prazo limite é atingido sem a coleta completa (RF17, Figura 53).

Quanto ao acompanhamento do processo, o usuário pode consultar a lista de documentos e o progresso de assinaturas (RF4, Figura 42). O sistema apresenta estados intermediários como "recebendo assinaturas", "incompleto" e "completo" (RF10, Figura 52). Ainda durante o recebimento de assinaturas, o sistema permite download do documento a qualquer momento, mas exibe um aviso quando o documento não está concluído (RF16, Figura 55).

9 CONCLUSÃO

Este trabalho apresentou a implementação da funcionalidade PdD no sistema ADES e a verificação do atendimento aos requisitos funcionais por rastreabilidade a evidências na interface, cumprindo os objetivos propostos na introdução. Por meio do levantamento de requisitos funcionais baseado em demandas levantadas no projeto e no trabalho de Lima (2023), foi possível definir um conjunto abrangente de funcionalidades que atendem aos casos de uso identificados, e que foram atendidos com sucesso. A implementação integrou-se à arquitetura de microserviços existente do ADES, com alterações no PAS, que serve o sistema do lado do navegador, e no GED, que lida com o ciclo de vida do documento do lado do servidor.

A implementação do PdD trouxe novos fluxos de realização e coleta de assinaturas digitais no sistema ADES, permitindo que usuários especifiquem signatários esperados, monitorem o progresso das assinaturas e recebam notificações automatizadas por e-mail e WhatsApp. Essas funcionalidades atendem diretamente às limitações identificadas no fluxo anterior do sistema, oferecendo maior controle e eficiência para quem prepara documentos, assina e coleta assinaturas.

9.1 TRABALHOS FUTUROS

Para trabalhos futuros, recomenda-se a realização de testes de usabilidade com públicos representativos dos perfis de uso do sistema, por exemplo, pessoas que frequentemente preparam documentos para coleta de assinaturas e pessoas que as assinam, de modo a verificar se a solução atende às demandas práticas desses grupos, identificar pontos de atrito na experiência e orientar melhorias iterativas na interface e nos fluxos.

Além disso, a partir de ideias geradas com a execução deste trabalho e conversas internas do projeto ADES sobre melhorias, sugere-se a exploração de funcionalidades que reduzam ainda mais a quantidade de passos necessários para os envolvidos no processo de assinatura. Uma possibilidade seria a implementação de preenchimento automático da descrição do documento utilizando técnicas de inteligência artificial, que poderia analisar o conteúdo do PDF e gerar automaticamente uma descrição contextual, reduzindo o esforço manual do usuário que prepara o documento e aumentando o entendimento de quem irá assinar em relação ao que será assinado. Para cobrir mais casos de uso do sistema, a ideia de especificar uma ordem para recebimento das assinaturas pode ser explorada, a fim de atender processos que requerem algum tipo de hierarquia para a realização de assinaturas. Por fim, para ampliar ainda mais os meios de contato com signatários, sugere-se a implementação de notificações via SMS, complementando ao e-mail e ao WhatsApp já disponíveis. Essas otimizações alinhariam-se ao objetivo de tornar o processo de assinatura digital cada vez mais fluido e eficiente, minimizando fricções na experiência do usuário e aumentando a adoção do sistema.

REFERÊNCIAS

- BRASIL. **Lei nº 14.063, de 23 de Setembro de 2020**. 2020. Diário Oficial da União. Disponível em: https://www.planalto.gov.br/ccivil_03/_ato2019-2022/2020/lei/l14063.htm.
- CABRAL, G. A. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação), **Assinatura de Documentos com Certificado de Uso Único e Chaves Geradas no Cliente**. Florianópolis, SC: [s.n.], 2023. 68 p. Disponível em: <https://repositorio.ufsc.br/handle/123456789/252306>.
- CODD, E. F. A relational model of data for large shared data banks. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 13, n. 6, p. 377–387, jun. 1970. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/362384.362685>.
- Docker Inc. **Docker Documentation**. 2025. Documentação oficial online. Plataforma de containerização para aplicações. Disponível em: <https://docs.docker.com/>.
- INTIC. **Assinador Avançado do INTIC**. 2023. Site institucional. Ferramenta de assinatura digital mantida pelo INTIC. Disponível em: <https://assinadoravancado.gov.mz/how>.
- JavaScript Team at Mozilla. **JavaScript Official Documentation**. 2025. Documentação oficial online. Página oficial da biblioteca React. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>.
- LabSEC. **ADES - ADvanced Electronic Signature**. 2023. Site institucional. Ferramenta de assinatura digital mantida pelo LabSEC. Disponível em: <https://ades.labsec.ufsc.br/how>.
- LIMA, G. M. Trabalho de Conclusão de Curso, **Assinador digital avançado: proposta de interface web de assinatura eletrônica acessada por smartphone**. Florianópolis: [s.n.], 2023. Disponível em: <https://repositorio.ufsc.br/handle/123456789/252306>.
- MAYR, L. et al. **One-Time Certificates for Reliable and Secure Document Signing**. 2024. Disponível em: <https://arxiv.org/abs/2208.03951>.
- Mozilla Developer Network. **CSS | MDN**. 2025. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/CSS>.
- Mozilla Developer Network. **HTML | MDN**. 2025. Documentação oficial da Mozilla Developer Network (MDN) sobre HTML. Disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTML>.
- Node.js Foundation. **Node.js Documentation**. 2025. Documentação oficial online. Ambiente de execução JavaScript para servidor. Disponível em: <https://nodejs.org/>.
- Pallets Team. **Flask Documentation (2.3.x)**. 2023. Documentação oficial online. Disponível em: <https://flask.palletsprojects.com/en/stable/>.
- PEROTTONI, E. et al. Menos certificação digital e mais identidade eletrônica: Icpedu e cafe em um assinador digital inclusivo. In: **Anais Estendidos do XXIII Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais**. Porto Alegre, RS, Brasil: SBC, 2023. p. 93–96. ISSN 0000-0000. Disponível em: https://sol.sbc.org.br/index.php/sbseg_estendido/article/view/27279.

PostgreSQL Global Development Group. **About PostgreSQL**. 2025. Documentação oficial online. Página institucional descrevendo o sistema de banco de dados PostgreSQL. Disponível em: <https://www.postgresql.org/about/>.

Python Software Foundation. **What is Python? The Executive Summary**. 2025. Documento online. Disponível em: <https://www.python.org/doc/essays/blurb/>.

React Team at Meta. **React Official Documentation**. 2025. Documentação oficial online. Página oficial da biblioteca React. Disponível em: <https://react.dev/>.

SCHARDONG, F.; MAYR, L.; CUSTÓDIO, R. **Processos de verificação e assinatura digital de documentos eletrônicos baseado em identidade eletrônica, certificado digital de uso único e blockchain**. 2022. Patente. BR102022012874. Concessão: 28 jun. 2022.

SCHMITZ, R. A. **Sistema de Gestão de Pessoas: especificação de requisitos funcionais para os processos seletivos simplificados da UFSC**. Dissertação (Dissertação (Mestrado Profissional em Administração Universitária)), Florianópolis, SC, 2018. Disponível em: <https://repositorio.ufsc.br/handle/123456789/191410>.

The Psycopg Team. **Psycopg Documentation**. 2025. Disponível em: <https://www.psycopg.org/docs/>.

TypeScript Team. **TypeScript Documentation**. 2025. Disponível em: <https://www.typescriptlang.org/docs/>.

Vite Team. **Vite Documentation**. 2025. Documentação oficial online. Ferramenta de build para aplicações frontend modernas. Disponível em: <https://vitejs.dev/>.

WhatsApp LLC. **Sobre o WhatsApp**. 2026. Site institucional. Disponível em: <https://www.whatsapp.com/about?lang=pt>.

WHITE, S. A. **Introduction to BPMN**. 2004. White Paper - Object Management Group (OMG). Disponível em: https://www.omg.org/bpmn/Documents/Introduction_to_BPMN.pdf.

World Bank Group. **Mozambique Overview**. 2025. Relatório institucional online. Página de resumo econômico e de desenvolvimento de Moçambique. Disponível em: <https://www.worldbank.org/en/country/mozambique/overview>.

APÊNDICE A – CÓDIGO-FONTE

O código-fonte desenvolvido neste trabalho está disponível no repositório institucional de códigos da Universidade Federal de Santa Catarina (UFSC), no endereço:

<https://codigos.ufsc.br/victor.sc/tcc-codigo-fonte>

Acesso em: 12 jul. 2026.

APÊNDICE B – ARTIGO PARA PUBLICAÇÃO

Desenvolvimento de um Sistema de Preparação de Documentos Eletrônicos para Gestão de Fluxos de Assinatura Digital

Victor dos Santos Camargo¹, Frederico Schardong², Ricardo Felipe Custódio¹

¹Laboratório de Segurança em Computação (LabSEC)
Universidade Federal de Santa Catarina (UFSC)
Florianópolis – SC – Brasil

²Instituto Federal do Rio Grande do Sul (IFRS)

victorscamargo@gmail.com

frede.sch@gmail.com

rfcustodio@gmail.com

Abstract. *This paper presents the Document Preparer (PdD), developed for the ADES digital signature ecosystem (UFSC). The system addresses the lack of control mechanisms in document workflows, enabling the registration of multiple signers, definition of deadlines, and automated notifications. The methodology encompassed requirement gathering, BPMN modeling, and implementation in the Signature Portal and Electronic Document Manager. Verification confirmed the complete fulfillment of the proposed requirements, optimizing the control over the electronic document workflow.*

Resumo. *Este artigo apresenta o Preparador de Documentos (PdD), desenvolvido para o ecossistema de assinaturas digitais ADES (UFSC). O sistema supre a carência de mecanismos de controle no fluxo documental, viabilizando o cadastro de múltiplos signatários, estipulação de prazos e automatização de notificações. A metodologia contemplou levantamento de requisitos, modelagem BPMN e implementação no Portal de Assinatura e no Gerenciador Eletrônico de Documentos. A verificação confirmou o atendimento integral aos requisitos propostos, otimizando o controle sobre o trâmite de documentos eletrônicos.*

1. Introdução

A sociedade encontra-se em uma era de crescente digitalização, onde processos físicos migram progressivamente para o meio eletrônico. A assinatura digital consolida-se como alternativa legal e eficaz no Brasil para operações jurídicas [BRASIL 2020], viabilizando essa transição. Nesse cenário, insere-se o projeto ADvanced Electronic Signature (ADES), desenvolvido pelo Laboratório de Segurança em Computação (LabSEC) da Universidade Federal de Santa Catarina (UFSC). O projeto fornece sistemas que permitem a assinatura, verificação e compartilhamento de documentos PDF de forma simplificada, apoiando-se unicamente na autenticação do usuário em um Provedor de Identidade (IdP) [LabSEC 2023]. Soluções derivadas do ADES impactam milhões de pessoas em

instituições como a Infraestrutura de Chaves Públicas para Ensino e Pesquisa (ICPEdu) [Perottoni et al. 2023] e o Instituto Nacional de Tecnologias de Informação e Comunicação (INTIC) de Moçambique [INTIC 2023].

Apesar de sua eficiência na assinatura de documentos isolados, o fluxo original do ADES apresentava limitações práticas ao atender casos de uso focados no trâmite de documentos e na coleta de assinaturas. Carecia-se de mecanismos de controle que permitissem especificar antecipadamente os signatários, estipular prazos limite e garantir o acompanhamento do status do documento. Sem essas ferramentas, o usuário que preparava o documento dependia de sistemas externos para notificar as partes e de monitoramento manual para confirmar a conclusão do processo. Visando sanar essas lacunas, surgiu a demanda pelo Preparador de Documentos (PdD), um conjunto de funcionalidades projetado para gerenciar e automatizar o ciclo de vida desses fluxos documentais [Lima 2023].

Diante desse contexto, este artigo descreve a implementação do PdD no ecossistema de assinaturas do projeto ADES e a verificação do atendimento aos seus requisitos funcionais. Para alcançar este propósito, a metodologia adotada englobou o levantamento de requisitos junto aos gestores do projeto, a proposição arquitetural com modelagem de processos em notação Business Process Model and Notation (BPMN), e a codificação nas camadas de *frontend* (Portal de Assinatura) e *backend* (Gerenciador Eletrônico de Documentos). Dessa forma, evidencia-se como a inserção desses novos mecanismos viabiliza o controle eficiente do trâmite de documentos eletrônicos, agregando valor e competitividade às soluções desenvolvidas pelo laboratório.

2. Fundamentação e Trabalhos Relacionados

O ecossistema do projeto ADES apoia-se na tecnologia de *One-Time Certificate* (OTC) para a realização das assinaturas digitais [Mayr et al. 2024, Schardong et al. 2022]. Diferente de Infraestruturas de Chaves Públicas (ICPs) tradicionais, o OTC dispensa o usuário de gerenciar chaves privadas ou lidar com revogação de certificados, delegando a comprovação de identidade a um Provedor de Identidade (IdP) e gerando um certificado de uso único para cada documento. Essa abordagem fundamenta a simplicidade do fluxo do ADES e de sistemas derivados, como o assinador da Rede Nacional de Ensino e Pesquisa (RNP) para a ICPEdu [Perottoni et al. 2023]. Como esses sistemas compartilham arquiteturas de microsserviços semelhantes, inovações concebidas no ADES possuem alto potencial de replicação e impacto em larga escala.

A evolução técnica do ADES tem sido objeto de pesquisas recentes que abriram caminho para o Preparador de Documentos (PdD). O trabalho de Cabral et al. [Cabral 2023], por exemplo, aprimorou a privacidade do sistema ao transferir a geração do par de chaves e a manipulação do PDF para o navegador do cliente, evitando o tráfego do documento não criptografado. Em suas conclusões, os autores apontaram como trabalho futuro a necessidade de um sistema de gerenciamento que permitisse convidar múltiplos usuários e acompanhar assinaturas pendentes. O PdD desenvolvido nesta pesquisa atende diretamente a essa lacuna, adicionando a camada de coordenação de fluxos colaborativos.

No que tange à interação do usuário, Manno Lima [Lima 2023] realizou testes

e propôs interfaces iniciais para o PdD, focadas em dispositivos móveis. Contudo, o protótipo não foi integrado à arquitetura do ADES e deixou pendências em relação ao uso em *desktop*, posicionamento de artefatos visuais no PDF e tratamento de estados intermediários (como documentos incompletos ou expirados). Este artigo avança sobre essa base, refinando os requisitos e implementando de fato o sistema.

Por fim, a complexidade de coordenar múltiplos atores e prazos exige uma modelagem clara, desafio semelhante ao enfrentado por Schmitz et al. [Schmitz 2018] ao mapear fluxos de gestão de pessoas na UFSC. Para transpor as regras de negócio institucionais para o software, o presente trabalho adotou a notação *Business Process Model and Notation* (BPMN) [White 2004], garantindo um alinhamento preciso entre os requisitos de trâmite documental e a implementação técnica.

3. Requisitos e Proposta de Arquitetura

O desenvolvimento do Preparador de Documentos (PdD) foi guiado por demandas de gestores do projeto ADES e por lacunas identificadas no fluxo de assinaturas original. Para garantir o controle eficiente sobre o trâmite documental, os requisitos funcionais (RFs) mais impactantes estabelecidos para o sistema incluíram:

- **Gestão de Signatários:** Capacidade de cadastrar os envolvidos especificando nome, e-mail, número de telefone e o nível de comprometimento em relação ao documento.
- **Controle de Prazos e Notificações:** Funcionalidade para definir uma data limite de coleta. O sistema deve disparar notificações automáticas por e-mail e WhatsApp solicitando as assinaturas, além de alertar o preparador sobre a conclusão ou expiração do prazo do lote.
- **Posicionamento Visual:** Permitir que o preparador defina, de forma interativa, as coordenadas exatas e a página do arquivo PDF onde o artefato visual de cada assinatura será ancorado.
- **Flexibilidade de Trâmite:** Suporte tanto a fluxos restritos (com signatários pré-definidos e opção de adiar a própria assinatura) quanto a fluxos abertos (compartilhamento de documento via *link* sem lista prévia de assinantes).

Para viabilizar essas funcionalidades, o PdD foi integrado à arquitetura de microsserviços do ADES [Perotoni et al. 2023], que é orquestrada por meio de contêineres *Docker*. A implementação concentrou-se em dois serviços fundamentais:

1. **Portal de Assinatura (PAS):** Interface web de interação do usuário (*frontend*), desenvolvida com *React* e *TypeScript*. O PAS recebeu as lógicas de roteamento e os componentes visuais para o fluxo de preparo, manipulação do PDF no navegador e os painéis de acompanhamento (*dashboard*) de status dos documentos.
2. **Gerenciador Eletrônico de Documentos (GED):** Microsserviço de *backend* implementado em *Python* e *Flask*, encarregado de armazenar e gerenciar o ciclo de vida dos arquivos. Sua lógica foi expandida para suportar a comunicação assíncrona de alertas, além da conexão com o banco de dados *PostgreSQL* para registrar as variáveis do preparo (prazos, signatários pendentes e coordenadas de assinatura).

A dinâmica de interação entre os usuários e a arquitetura do sistema foi modelada utilizando a notação BPMN para garantir alinhamento com as regras de negócio institucionais. O fluxo principal, ilustrado de forma simplificada na Figura 1, estrutura-se em três macroetapas: (i) o preparo do documento, onde o usuário insere o arquivo e configura as restrições no PAS; (ii) a coleta, onde o GED orquestra as notificações e aguarda as assinaturas; e (iii) o recebimento, onde o documento finalizado ou expirado retorna ao preparador.

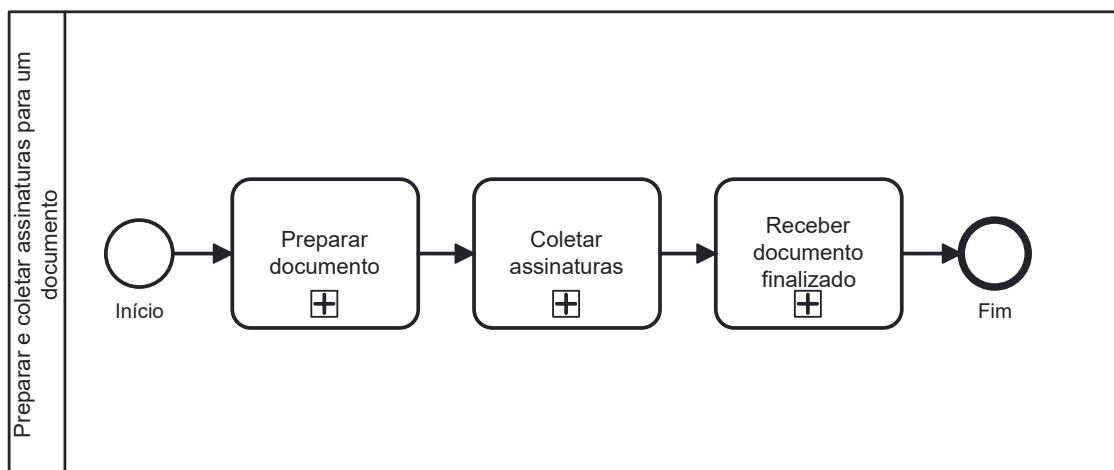


Figura 1. Processo principal de preparo de documentos e coleta de assinaturas.

4. Implementação do Preparador de Documentos

A materialização do Preparador de Documentos exigiu intervenções estruturais em ambas as extremidades do ecossistema ADES. No *frontend* (PAS), o código-fonte em *React* e *TypeScript* foi acrescido de um módulo dedicado, centralizando componentes de *dashboard* para listagem de documentos, interfaces modulares de cadastro de signatários e um visualizador de PDF interativo. Este último foi crucial para permitir a captura precisa das coordenadas de tela (x e y) onde o artefato de assinatura deve ser ancorado.

Contudo, as modificações mais profundas ocorreram no *backend* (GED) e em sua modelagem relacional no *PostgreSQL*. Originalmente, o sistema estruturava-se em torno de uma tabela de lotes (*batches*) orientada apenas à permissão básica de acesso pelo proprietário. Para suportar o trâmite colaborativo, a tabela *batches* foi expandida para registrar metadados complexos, como o status lógico de conclusão (*completed*), data de finalização, indicadores de acessibilidade por *link* e informações de contato do preparador.

Para gerenciar os múltiplos atores do processo, projetou-se a tabela relacional *batch_signers*. Essa estrutura tornou-se o pilar do monitoramento de progresso, armazenando não apenas a identidade e o contato dos signatários, mas também seus respectivos níveis de comprometimento legal, a página e as coordenadas do carimbo visual no documento, e o registro do exato momento da assinatura (*signed_at*). A ausência de valor neste último campo atua como o principal indicador de assinatura pendente.

A orquestração dessa nova base de dados foi implementada na aplicação em

Python/Flask por meio de três categorias de funções lógicas:

- **Gestão de Ciclo de Vida e Estados:** Foram criadas rotinas assíncronas (como `complete_expired_private_prepared_batches`) para varrer a base de dados periodicamente, identificando e encerrando lotes que atingiram a data limite, além de expurgar conteúdos binários antigos para otimização de armazenamento.
- **Motor de Notificações:** A lógica de comunicação passou a centralizar o disparo de alertas automáticos com o envio de e-mails detalhados e integrações com o WhatsApp (quando o telefone é cadastrado), notificando tanto a requisição de novas assinaturas quanto avisos de lotes concluídos ou expirados.
- **Controle de Acesso e Registro:** Novas camadas de verificação de permissões (ex.: `is_user_expected_signer`) foram adicionadas para garantir que, no fluxo de signatários pré-especificados, apenas os usuários autenticados via IdP correspondentes aos e-mails cadastrados possam visualizar e assinar o documento correspondente.

5. Resultados e Verificação

A verificação do Preparador de Documentos foi conduzida por meio da execução de cenários de uso reais no sistema integrado, confrontando o atendimento aos requisitos funcionais (RF) mapeados. O cenário principal englobou o preparo de um documento com múltiplos signatários, o disparo de notificações e o monitoramento até a sua conclusão.

Durante a etapa de preparo, confirmou-se a eficácia da interface interativa: o usuário foi capaz de carregar um arquivo PDF, estabelecer uma data limite para o trâmite e cadastrar os signatários informando dados de identificação, telefone e nível de comprometimento legal. A Figura 2 ilustra a consolidação dessa etapa, demonstrando a funcionalidade de posicionamento visual, na qual o preparador ancora com precisão o artefato de assinatura de cada participante sobre o documento original. Verificou-se, ainda, que o sistema permite ao preparador incluir a si próprio como signatário com apenas um clique ou, se preferir, adiar sua assinatura para um momento posterior.

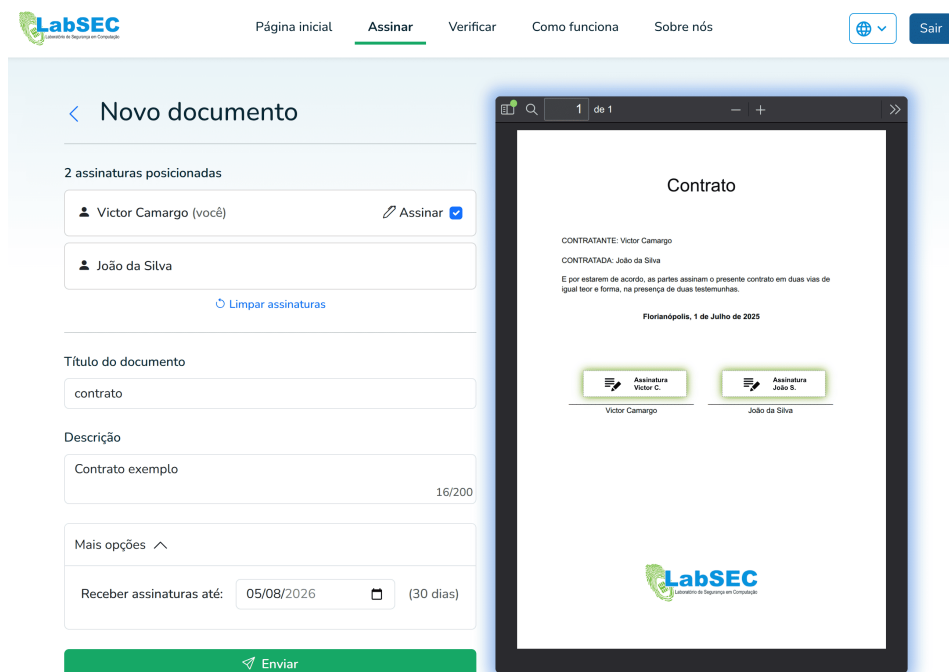


Figura 2. Tela de preparo demonstrando a marcação visual de múltiplos signatários sobre o documento.

No que tange à coleta de assinaturas, os testes confirmaram o funcionamento do motor de notificações. O sistema disparou automaticamente avisos por e-mail e mensagens estruturadas via WhatsApp contendo os *links* de acesso e o contexto da assinatura (Figura 3).

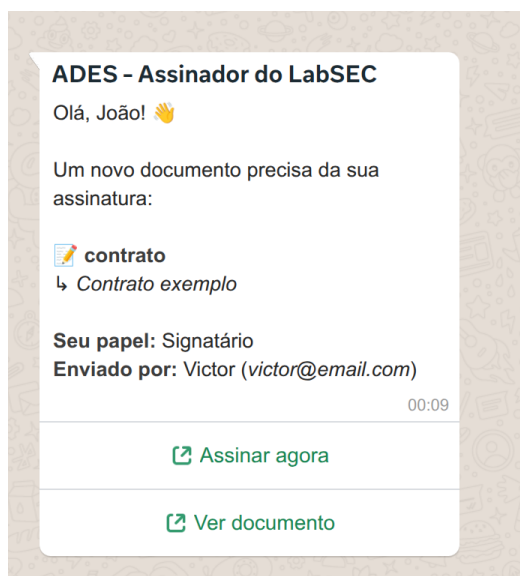


Figura 3. Mensagem de pedido de assinatura recebida pelo signatário no WhatsApp.

A verificação do controle de ciclo de vida e monitoramento foi evidenciada por meio do *dashboard* de acompanhamento, ilustrado na Figura 4. O sistema demonstrou

capacidade de gerenciar e exibir corretamente os estados intermediários do processo documental: lotes aguardando assinaturas (pendentes), lotes expirados sem sucesso (incompletos) e lotes finalizados (completos). Além de atualizar o painel visual, o sistema notificou o preparador automaticamente nos cenários de sucesso ou de expiração do prazo, cumprindo as premissas de rastreabilidade do processo.

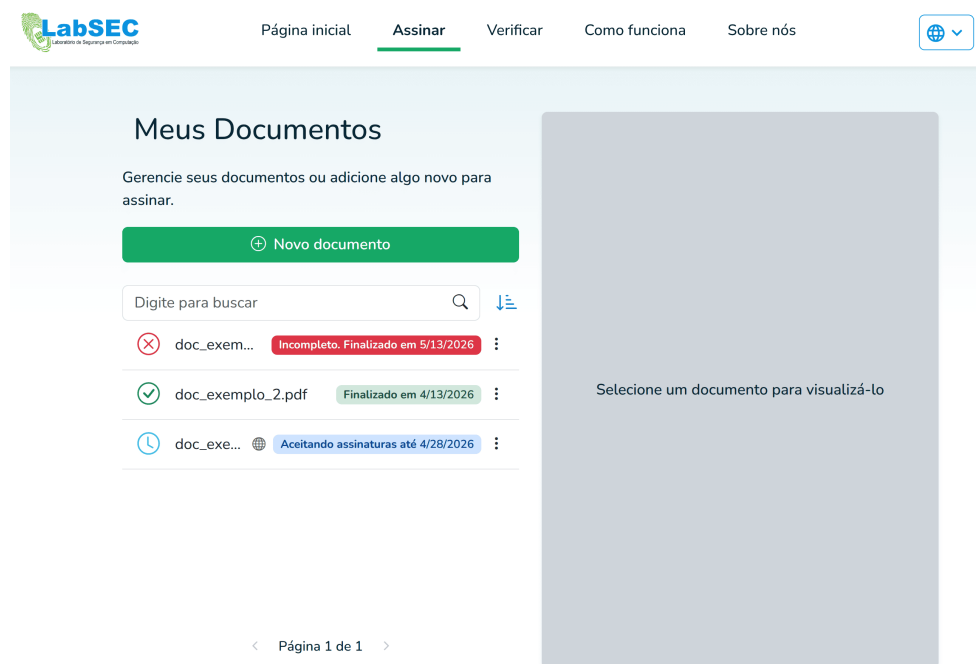


Figura 4. Painel de monitoramento apresentando documentos em distintos estados de completude.

Adicionalmente, os fluxos alternativos foram testados com êxito, incluindo a viabilidade de compartilhamento público de documentos via *link* (sem signatários pré-especificados) e a permissão para que o usuário realize o *download* de um documento incompleto de forma emergencial, desde que mediante alerta prévio do sistema sobre a situação pendente.

6. Conclusão e Trabalhos Futuros

Este artigo apresentou a concepção, implementação e verificação do Preparador de Documentos (PdD) no ecossistema de assinaturas digitais ADES. A solução desenvolvida superou as limitações do fluxo original ao introduzir mecanismos de controle sobre o trâmite documental, permitindo a especificação de signatários esperados, o monitoramento do progresso das assinaturas e o disparo automatizado de notificações multiplataforma (e-mail e WhatsApp). A integração dessas funcionalidades à arquitetura de microsserviços existente, abrangendo o Portal de Assinatura no navegador e o Gerenciador Eletrônico de Documentos no servidor, ampliou o conjunto de recursos disponíveis para quem prepara documentos e coordena a coleta de assinaturas no ADES.

Como trabalhos futuros, vislumbra-se a condução de testes de usabilidade com amostras representativas do público-alvo (preparadores e signatários frequentes), com o

intuito de refinar a experiência de uso e identificar pontos de atrito na interface. Sob a ótica de evolução funcional, propõe-se o uso de técnicas de inteligência artificial para analisar o conteúdo do documento PDF e gerar descrições contextuais de forma automática, otimizando o esforço de preparo. Por fim, a implementação de notificações via SMS e a adição de um recurso para definir a ordem hierárquica (sequencial) de assinaturas são apontadas como expansões promissoras para cobrir cenários administrativos ainda mais complexos.

Referências

- BRASIL (2020). Lei nº 14.063, de 23 de setembro de 2020. Diário Oficial da União.
- Cabral, G. A. (2023). Assinatura de documentos com certificado de uso Único e chaves geradas no cliente.
- INTIC (2023). Assinador avançado do intic. Site institucional. Ferramenta de assinatura digital mantida pelo INTIC.
- LabSEC (2023). Ades - advanced electronic signature. Site institucional. Ferramenta de assinatura digital mantida pelo LabSEC.
- Lima, G. M. (2023). Assinador digital avançado: proposta de interface web de assinatura eletrônica acessada por smartphone.
- Mayr, L., Zambonin, G., Schardong, F., and Custódio, R. (2024). One-time certificates for reliable and secure document signing.
- Perottoni, E., Costa, B., Müller, F., Camargo, V., Schardong, F., Silvano, W., Mayr, L., Custódio, R., Rocha, L., Lyra, C., Matushima, R., and Rieckmann, N. (2023). Menos certificação digital e mais identidade eletrônica: Icpedu e cafe em um assinador digital inclusivo. In *Anais Estendidos do XXIII Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais*, pages 93–96, Porto Alegre, RS, Brasil. SBC.
- Schardong, F., Mayr, L., and Custódio, R. (2022). Processos de verificação e assinatura digital de documentos eletrônicos baseado em identidade eletrônica, certificado digital de uso único e blockchain. Patente. BR102022012874. Concessão: 28 jun. 2022.
- Schmitz, R. A. (2018). Sistema de gestão de pessoas: especificação de requisitos funcionais para os processos seletivos simplificados da ufsc. Dissertação (mestrado profissional em administração universitária), Florianópolis, SC.
- White, S. A. (2004). Introduction to BPMN. White Paper - Object Management Group (OMG).