

# Um sistema web para auxiliar soluções na conformidade com a LGPD

Matheus Frej

Universidade Federal de Pernambuco  
Recife, Brasil  
mflc@cin.ufpe.br

Waldemar Ferreira

Universidade Federal Rural de Pernambuco  
Recife, Brasil  
waldemar.ferreira@ufrpe.br

Ivonildo Pereira

Universidade Federal de Pernambuco  
Recife, Brasil  
ipgn@cin.ufpe.br

Sergio Soares

Universidade Federal de Pernambuco  
Recife, Brasil  
scbs@cin.ufpe.br

## ABSTRACT

Compliance with the General Data Protection Law (LGPD) is essential for Brazilian companies in protecting personal data. This study proposes the enhancement of an LGPD compliance checklist by transforming it into a web tool to facilitate the evaluation and implementation of legal requirements. The development of the tool included analyzing feedback from experts, notably the expert who assessed the inspirational checklist. The process also involved gathering requirements, defining improvement criteria, developing, and testing. The tool's evaluation was conducted on real projects by companies involved in technological innovation. The evaluation results revealed significant efficiency in identifying and addressing compliance gaps, highlighting the tool's potential to improve LGPD compliance.

## RESUMO

A conformidade com a Lei Geral de Proteção de Dados (LGPD) é de grande importância para as empresas brasileiras, visando garantir a proteção dos dados pessoais. Este trabalho propõe a melhoria de um checklist de conformidade com a LGPD, transformando-o em uma ferramenta web para facilitar a avaliação e a implementação das exigências legais. A construção da ferramenta incluiu análise do feedback de especialistas, com destaque para o especialista que avaliou o checklist inspirador. Além disso, o processo envolveu levantamento de requisitos, definição de critérios de melhoria, desenvolvimento e testes. A avaliação da ferramenta foi conduzida em projetos reais de empresas ligadas à inovação tecnológica. Os resultados da avaliação revelaram uma eficiência significativa na identificação e correção de lacunas de conformidade, destacando o potencial da ferramenta para melhorar a conformidade com a LGPD.

Apresentação: <https://zenodo.org/records/11390727>

## KEYWORDS

LGPD, Conformidade, Ferramenta web, Proteção de dados

## 1 INTRODUÇÃO

Com o avanço acelerado da tecnologia nos últimos anos, a coleta e o processamento de dados pessoais na internet tornaram-se atividades fáceis, comuns e essenciais para empresas e instituições [12]. Esses dados, que fornecem informações valiosas para a tomada de

decisões estratégicas e a evolução dos sistemas, também trazem à tona questões críticas de privacidade e segurança. A manipulação inadequada ou a exposição indevida de dados pessoais e sensíveis pode resultar em graves violações da privacidade, afetando tanto indivíduos quanto organizações [1]. Tendo em vista esse problema, foi promulgada na Europa a General Data Protection Regulation (GDPR) em 2016 [16], sendo uma lei pioneira na regulamentação do tratamento de dados pessoais e servindo como base para a lei brasileira de mesmo tema, a Lei Geral de Proteção de Dados (LGPD) [6]. Além dessas duas leis relevantes, atualmente mais de 120 países já possuem leis regulatórias nesse âmbito [40].

Diante desse cenário regulatório complexo, os desenvolvedores de software enfrentam um desafio significativo para garantir que suas soluções estejam em conformidade com a legislação vigente. Muitos não sabem precisamente quais atributos são necessários no software para que ele esteja em conformidade com a lei [27]. Para abordar essa dificuldade, Mendes [27] propôs uma ferramenta de avaliação que consiste em uma checklist destinada à inspeção de sistemas computacionais, em formato de planilha Excel. O objetivo dessa checklist é verificar se os sistemas estão em conformidade com a LGPD e se seguem práticas recomendadas de tratamento de dados.

Após isso, Pereira *et al.* [31] expandiram a aplicabilidade da checklist para avaliar o tratamento e segurança de dados em dispositivos IoT (Internet das Coisas). Embora as questões relacionadas a esses dispositivos sejam opcionais na checklist, sua inclusão é crucial para garantir a segurança dos dados em ambientes de IoT. Tanto Mendes [27] quanto Pereira *et al.* [31] sugerem melhorias futuras para a checklist, como automatizá-la para validação de campos, expandir sua aplicabilidade para outras áreas além de IoT e atualizá-la conforme novas leis são criadas. Pereira *et al.* [31] também mencionam o desenvolvimento de uma ferramenta web para facilitar o acesso e utilização da checklist pelos inspetores de sistemas.

Considerando as demandas identificadas por Mendes [27] e Pereira *et al.* [31], surge a oportunidade de desenvolver um sistema web que integre a checklist LGPD estendida com IoT, oferecendo funcionalidades complementares. O foco desse sistema será na facilidade de uso, na intuitividade para o usuário inspetor de sistema e na abordagem de lacunas que a planilha Excel, devido às suas limitações inerentes, não consegue solucionar.

O artigo está estruturado da seguinte forma: A seção 2 apresenta a fundamentação teórica que serve de base para este trabalho. A

seção 3 discute os trabalhos relacionados, oferecendo uma visão ampla do contexto atual. A seção 4 oferece uma visão preliminar do checklist inicial de conformidade com a LGPD. Na seção 5, detalha-se o desenvolvimento da ferramenta proposta, desde a concepção do protótipo até a implementação final do sistema. A seção 6 discute a avaliação da ferramenta. Por fim, a seção 7 conclui o artigo, resumindo as principais descobertas e sugerindo direções futuras para pesquisas.

## 2 FUNDAMENTAÇÃO TEÓRICA

Esta seção resume a LGPD e a segurança em sistemas de informação empresariais, abordando a literatura relevante e destacando aspectos essenciais para o entendimento do conteúdo deste artigo.

### 2.1 Surgimento da LGPD

A proliferação do modelo de negócios digital, a partir da década de 90, foi o principal impulsionador para o surgimento de regulamentações de proteção de dados. O aumento da coleta e processamento de informações pessoais levantou preocupações sobre privacidade, levando governos a criarem legislações como a GDPR na União Europeia e a LGPD no Brasil [33]. Quando se fala de leis regulatórias de dados, a mais relevante que vem à tona é a GDPR. Ela começou a ser discutida em 2012, foi promulgada em 2016 e começou a ser aplicada no dia 25 de maio de 2018 [16]. Ela foi um marco muito importante para a Europa, até mesmo economicamente, pois a União Europeia passou a exigir que países que fossem se relacionar com ela também deveriam ter regulamentações parecidas em relação a dados [36]. Esse foi mais um fator que impulsionou o Brasil a criar a sua própria lei de regulamentação de tratamento de dados. A lei brasileira foi aprovada em agosto de 2018, mas só começou a ser aplicada dois anos depois, em agosto de 2020. Esse intervalo de tempo se deve ao fato de que as empresas precisaram entender a lei e preparar os seus sistemas para estarem conformes com a LGPD. A lei abrange qualquer um que possa tratar dados pessoais ou sensíveis, seja pessoa física, pessoa jurídica, empresa privada e empresa pública [30].

### 2.2 Segurança em Sistemas de Informação e Conformidade com a LGPD nas Empresas

De acordo com Batista [3], um sistema de informação é um conjunto de componentes interligados que coletam, processam, armazenam e distribuem informações para apoiar a tomada de decisões, coordenação e controle em uma organização. Esses sistemas podem conter informações sobre pessoas, locais e itens importantes e podem ser tanto computadorizados quanto físicos e manuais.

A segurança em sistemas de informação pode ser vista de vários ângulos. Um exemplo de problema é a infecção por software malicioso. Apesar dos esforços contínuos dos profissionais de segurança cibernética, os invasores sempre procuram novas vulnerabilidades [23]. Incidentes de segurança afetam diversas empresas e instituições. Em 2019, dados de mais de 60 instituições de ensino foram comprometidos por vulnerabilidades em um único sistema [25].

Diante do risco de vazamento de dados pessoais, as empresas estão intensificando seus esforços para garantir a conformidade com leis de proteção de dados [13]. Obter essa conformidade apresenta

muitos desafios. Um dos principais deles é traduzir os aspectos legais em medidas técnicas viáveis para implementação [15, 24, 37, 41].

Para enfrentar esses desafios, as empresas precisam adotar uma abordagem abrangente de segurança da informação, que inclua prevenção, detecção e recuperação. Prevenção envolve proteger hardware, arquivos, dados e o perímetro da rede com ferramentas como firewalls. A detecção requer sistemas de alerta que notificam sobre invasões. A recuperação foca em criar cópias de segurança dos dados e ter hardwares de reserva [32]. Além disso, as empresas devem implementar medidas eficazes para proteger os dados pessoais, garantindo conformidade legal e evitando penalidades.

## 3 TRABALHOS RELACIONADOS

O estudo de Mendes [27] criou uma checklist para avaliar a conformidade de softwares com a LGPD, preenchendo uma lacuna na pesquisa que correlaciona sistemas com a legislação. Inspirado pela GDPR, ele desenvolveu essa ferramenta pioneira, demonstrando sua eficácia ao aplicá-la em um sistema universitário. Pereira *et al.* [31] expandiram essa checklist para incluir sistemas IoT, com resultados positivos em testes industriais. Nosso artigo busca integrar essas iniciativas em uma única ferramenta, ampliando o suporte para a conformidade de diversos sistemas com a LGPD.

Ainda no campo de checklists de inspeção, Cerqueira, Mello e Travassos [9] desenvolveram a técnica LGPDCheck. Esta ferramenta foi criada por meio de várias etapas, incluindo entrevistas com especialistas, revisão bibliográfica, análise da LGPD, construção dos artefatos e sua avaliação. O checklist foi projetado para três níveis de inspeção: requisitos do sistema (verificação de documentos de requisitos), sistemas em execução (verificação durante a execução dos sistemas) e práticas organizacionais (investigação das práticas relacionadas à segurança e responsabilidade). Os resultados mostraram que o checklist foi eficaz na identificação de defeitos relacionados aos princípios da LGPD em artefatos de software. A avaliação experimental com três equipes de desenvolvimento em um sistema IoT real demonstrou a viabilidade e utilidade da LGPDCheck. Nossa ferramenta se diferencia por se concentrar em facilitar o uso e o acesso do checklist através de uma plataforma web com uma interface intuitiva e funcionalidades complementares que auxiliam a inspeção.

Por fim, Cejas *et al.* [8] apresentam uma abordagem inovadora que automatiza a verificação de conformidade de Acordos de Processamento de Dados (DPAs) em relação à GDPR. DPAs são contratos estabelecidos entre controladores e operadores de dados que ditam os termos e condições para o processamento de dados pessoais. Embora o trabalho se concentre na GDPR, o objetivo é relevante e se encaixa no escopo desta seção. Na pesquisa, foram empregadas técnicas de Processamento de Linguagem Natural (NLP) para identificar a conformidade dos DPAs com a lei europeia. A ferramenta desenvolvida alcançou uma acurácia média de 84%. Em comparação com outras pesquisas existentes que utilizam ferramentas NLP prontas para uso, a abordagem proposta obteve um ganho de 20% de acurácia, demonstrando uma melhoria considerável. A ferramenta de NLP realiza uma inspeção apenas dos textos dos DPAs, enquanto a nossa ferramenta permite uma inspeção completa de todo o sistema.

#### 4 VISÃO PRELIMINAR DO CHECKLIST INICIAL DE CONFORMIDADE COM A LGPD

Mendes, Viana e Rivero [26] criaram um checklist inicial para avaliar a conformidade de sistemas com a LGPD, usando uma Revisão Sistemática da Literatura (RSL) e análises comparativas entre a LGPD e a GDPR. A versão inicial em PDF continha 52 itens distribuídos em quatro colunas, com identificadores, descrições e fontes dos itens.

Posteriormente, o checklist foi aprimorado, resultando em uma planilha com 47 itens obrigatórios da LGPD e nove itens recomendados, além de 27 itens adicionais para soluções de IoT. Esta versão inclui grau de severidade (leve, grave ou catastrófico) e recomendações para resolver inconformidades [31].

Foi conduzida uma análise minuciosa da checklist existente, examinando seu funcionamento geral, suas funcionalidades essenciais e identificando elementos que poderiam ser reaproveitados, aprimorados ou incrementados. A Figura 1 oferece uma visualização da checklist original.



Figure 1: Visão da primeira versão do checklist no formato Excel.

Após essa análise, tornou-se evidente que a checklist atual não atende completamente às necessidades, e que um sistema web seria mais adequado por várias razões:

- A personalização do preenchimento, validação e geração de gráficos é mais viável em um sistema web.
- Um sistema web é significativamente mais acessível do que uma planilha, pois pode ser acessado de qualquer navegador.
- O controle de versão não pode ser efetivamente implementado em uma planilha.
- A manutenção e escalabilidade de um sistema web oferecem vantagens significativas em comparação com uma planilha.

No estudo conduzido por Pereira *et al.* [31], os participantes identificaram uma área principal de aprimoramento: a necessidade de tornar obrigatórios os comentários sempre que um avaliador considerar um item não conforme com o sistema. Esta medida visa garantir que os motivos por trás de uma marcação sejam claramente comunicados, evitando ambiguidades decorrentes da falta de explicações no checklist anterior. Com base nesse apontamento, torna-se evidente a necessidade de desenvolver um sistema que mantenha os conteúdos e funcionalidades essenciais da planilha, ao mesmo tempo em que incorpora as melhorias sugeridas. Além disso, serão consideradas as sugestões de trabalhos futuros de Mendes [27] e Pereira *et al.* [31].

Ademais, foram conduzidas reuniões com os desenvolvedores do checklist da primeira versão, a fim de compreender integralmente sua operacionalização, identificar suas funcionalidades fundamentais e explorar oportunidades de reutilização, aprimoramento e expansão.

#### 5 DESENVOLVIMENTO DA SOLUÇÃO

Esta seção detalha o desenvolvimento da solução, incluindo a criação do protótipo, a implementação inicial do *frontend* com React e TypeScript, a jornada do usuário, e a evolução do sistema com a adição de um *backend* usando Express e Prisma para maior funcionalidade e segurança.

##### 5.1 Protótipo

A prototipagem é um processo no desenvolvimento de sistemas que envolve criar um modelo preliminar, permitindo ajustes antes da implementação. Este método gera ideias sobre o design da interface e avalia a qualidade da solução em um estágio inicial [4].

Em sistemas web, a prototipagem desempenha um papel fundamental. Além de facilitar a elicitação de requisitos, ela define a aparência visual da interface do usuário, fornecendo uma base essencial para o desenvolvimento [18]. Neste trabalho, a checklist original atua como protótipo, pois contém todos os elementos necessários para a construção do sistema web. Conforme ilustrado na Figura 2, a planilha possui abas, cada uma representando uma etapa distinta, o que permitiu o mapeamento dessas abas para páginas do sistema.

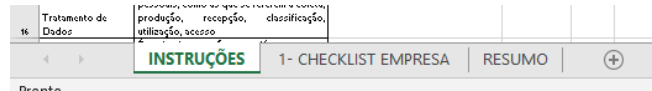


Figure 2: Abas da planilha do checklist.

##### 5.2 Ideia Inicial de Desenvolvimento

Na análise inicial deste estudo de desenvolvimento, foi determinado que não seria necessário implementar persistência de dados do usuário. A checklist opera eficientemente em um formato de planilha, dispensando a necessidade de armazenamento centralizado dos dados.

Considerando esses aspectos, iniciamos o desenvolvimento do *frontend* do sistema web. *Frontend* é encarregado de apresentar a interface ao usuário e possibilitar sua interação com o sistema [17]. Nesta parte, optamos pelo framework React<sup>1</sup> como principal recurso de desenvolvimento, dado seu amplo uso e reconhecimento na comunidade de desenvolvimento *frontend*, como observado em estudos anteriores [22, 29, 43]. Sua popularidade se reflete em uma comunidade ativa e vasto conteúdo online para suporte. O React, fundamentado na modularização e reutilização de código por meio de componentes, prioriza a eficiência e manutenibilidade do código [5]. Além disso, assim como outros frameworks de ponta no mercado, destaca-se pela reatividade, possibilitando a atualização automática

<sup>1</sup><https://react.dev/>

da interface conforme as variáveis reativas são modificadas, resultando em uma experiência do usuário fluida e dinâmica.

O React, concebido para interoperar com JavaScript, também se integra notavelmente com TypeScript. Optamos por TypeScript devido à sua forte tipagem, que aumenta a segurança para os desenvolvedores e para o código. Além disso, o sistema foi projetado para ser responsivo, adaptando-se a diferentes tamanhos de tela. A tipagem estática do TypeScript ajuda a detectar inconsistências durante o desenvolvimento, facilitando a manutenção e escalabilidade do projeto

### 5.3 Jornada do Usuário na Checklist LGPD

Ao abrir o sistema web, o usuário é direcionado inicialmente à página de instruções, conforme ilustrado na Figura 3. Esta página foi especificamente adaptada para o sistema, oferecendo orientações claras para o usuário. Na mesma figura, observa-se a presença de um cabeçalho, um componente comum em sistemas web, que contém o nome do sistema e um botão para alternar instantaneamente entre os temas claro e escuro.

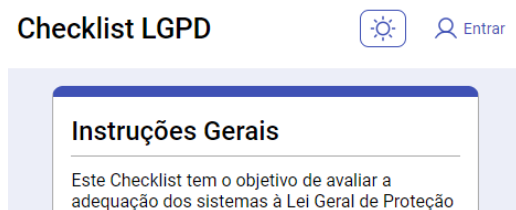


Figure 3: Página de instruções no sistema.

Na sequência do fluxo do usuário, a próxima tela permite escolher quais famílias de checklists serão avaliadas na inspeção do sistema atual. Esta funcionalidade atende a uma das sugestões para trabalhos futuros propostas por Pereira *et al.* [31]. Dessa forma, a checklist adapta-se às necessidades específicas do sistema avaliado, filtrando os itens com base na escolha do usuário. O resultado da implementação desta tela pode ser observado na Figura 4.

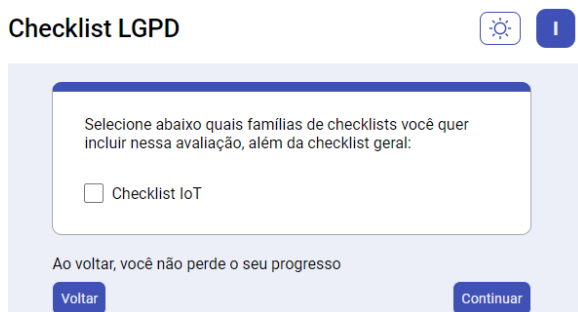


Figure 4: Seleção de famílias de checklists para inspeção.

Após esta etapa, o usuário da ferramenta encontrará a primeira tela de itens a serem respondidos. Na checklist original, tanto os itens obrigatórios quanto os não obrigatórios estão na mesma aba. No entanto, constatou-se que separá-los em duas páginas distintas

no sistema web é mais eficiente, pois divide melhor as etapas e reduz o cansaço do inspetor.

Na checklist original, o usuário responde aos itens marcando "Sim", "Não" ou "Não se aplica". Durante os testes realizados por Pereira *et al.* [31], concluiu-se que seria benéfico obrigar o inspetor a preencher os campos "Grau de Severidade" e "Comentário do Avaliador" caso a resposta seja "Não". Essa regra foi implementada no desenvolvimento do sistema, e o usuário só pode avançar se cumprir essa exigência.

Além disso, Mendes [27] sugere, como trabalho futuro, a possibilidade de filtrar a visualização da checklist por tipo de resposta, como mostrar apenas os itens respondidos com "Não". Esta funcionalidade foi incorporada como um requisito do sistema.

A próxima tela no fluxo do usuário trata dos itens não obrigatórios. Um destaque desta tela é a inclusão de itens relacionados à IoT [31], caso o inspetor tenha selecionado essa opção. Após essa parte, chega-se à etapa final do processo de inspeção do sistema, a tela de "Relatório". Nessa tela, o usuário pode visualizar gráficos que mostram as estatísticas de preenchimento dos itens obrigatórios e não obrigatórios de maneira centralizada. A Figura 5 apresenta um exemplo dos gráficos. Abaixo deles, todos os itens da checklist são exibidos. Uma nova funcionalidade desta etapa, em relação à planilha, é a geração de relatórios em PDF.

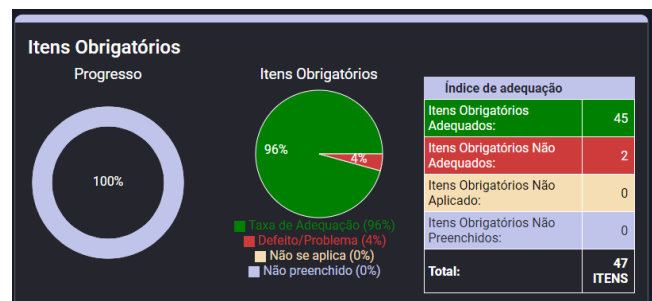


Figure 5: Exemplo do gráfico.

Ao concluir a primeira versão, verificou-se que a implementação do sistema web apenas com *frontend* limitava a presença de funcionalidades promissoras, que seriam diferenciais em comparação com a planilha. Considerando essa limitação, iniciou-se o desenvolvimento da segunda versão da ferramenta, agora incluindo um *backend* com banco de dados. O *backend* é o núcleo de uma aplicação, abrangendo o servidor e suas operações. Ele lida com o armazenamento, manipulação de dados e execução das lógicas de negócio, funcionando sem contato direto com os usuários [21].

### 5.4 Evolução da Ferramenta

Para desenvolver o *backend*, é essencial inicialmente definir a estrutura do banco de dados, uma vez que ele determina os atributos das entidades e suas interações. A abordagem mais comum para essa estruturação é a utilização do modelo relacional [10]. A Figura 6 apresenta o modelo da ferramenta.

O modelo apresentado inclui três tabelas principais. A tabela de usuários armazena informações sobre os usuários cadastrados na

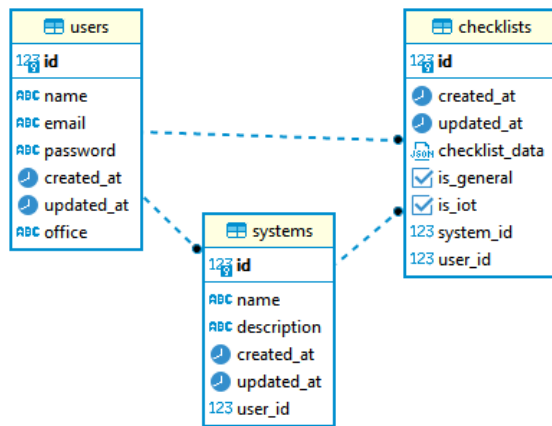


Figure 6: Modelo relacional da ferramenta.

ferramenta, como nome e cargo. A tabela de sistemas contém os atributos nome e descrição do sistema, sendo que cada sistema está associado a um usuário. A tabela de checklists registra avaliações, associando cada avaliação a um usuário e a um sistema. Com a definição do banco de dados concluída, foi possível prosseguir para o desenvolvimento do *backend*.

Para o *backend*, foi utilizada a linguagem TypeScript [28]. Ela é uma linguagem em constante evolução que se baseia em JavaScript, adicionando recursos de verificação de tipo estático [38]. Por ser a mesma empregada no *frontend*, a manutenção da ferramenta é facilitada, pois um desenvolvedor precisa conhecer apenas essa linguagem para entender ambas as partes do sistema. Considerando a natureza do *backend*, a utilização de um framework que gerencie requisições é benéfica. O framework escolhido foi o Express, devido à sua simplicidade, velocidade e escalabilidade.

Além disso, para comunicar o *backend* com o banco de dados, utilizou-se uma ORM (Object Relational Mapper) [14]. A ORM escolhida foi o Prisma [34]. Ele simplifica o gerenciamento do banco de dados, lidando bem com migrações do banco e da comunicação com ele. Com o *backend* estruturado, ele começou a ser codificado.

Durante o desenvolvimento da ferramenta, o tratamento correto dos dados foi uma prioridade constante. Boas práticas foram implementadas no *backend* para garantir a segurança dos dados pessoais dos usuários. Uma dessas práticas foi a criptografia das senhas ao armazená-las no banco de dados. Além disso, utilizou-se JWT (JSON Web Token) [19] para a autenticação dos usuários. O JWT é um tipo de *token* que contém informações essenciais e é protegido por uma assinatura digital, gerada com uma chave secreta [2]. Essa chave é uma variável de ambiente que permanece exclusivamente no servidor e não é enviada para o sistema de versionamento do código, garantindo maior segurança. Após o login, o servidor envia um JWT de autenticação ao *frontend*. Esse JWT é anexado ao cabeçalho de todas as requisições HTTP feitas pelo *frontend* ao servidor. Se o JWT estiver ausente, corrompido ou inválido, as requisições que requerem autenticação não serão processadas pelo *backend*. Além disso, mesmo que um usuário mal-intencionado consiga o *token* de outra pessoa, ele será inútil após o prazo de expiração.

Com a integração do banco de dados e do *backend*, diversas funcionalidades importantes se tornam disponíveis. O usuário pode cadastrar sistemas e preencher checklists associadas a esses sistemas. Além disso, o usuário tem a capacidade de visualizar uma listagem das checklists já preenchidas, bem como editá-las e excluí-las.

Por fim, foi necessário adaptar o código do *frontend* para se conectar ao *backend* e criar novas telas. É importante mencionar que, embora o usuário possa fazer login, ele também pode realizar avaliações de sistemas sem ter uma conta. No entanto, sem autenticação, o usuário não terá o histórico salvo e não terá acesso a todas as funcionalidades disponíveis apenas para usuários autenticados.

## 5.5 Resultado da Solução

Após a conclusão do desenvolvimento, a ferramenta foi disponibilizada na internet para testes de usuários em um processo conhecido como *deploy*. Considerando que o sistema web é composto por dois projetos interdependentes, o *backend* e o *frontend*, ambos necessitaram de *deploy* individual. Além disso, o banco de dados, inicialmente executado localmente, também precisou ser disponibilizado remotamente.

Para o *deploy* do *frontend*, foi utilizada a plataforma Vercel [42], um serviço de nuvem focado em *frontend*. Uma das principais vantagens do Vercel é seu sistema de entrega contínua [39], que automatiza a publicação de novas versões. Sempre que um novo *commit* é feito na *branch* principal, a plataforma executa o *pipeline* de publicação, disponibilizando a nova versão do site. O Vercel também gera um *link* único para acesso ao site.

Para o *deploy* do *backend* e do banco de dados, foi escolhida a plataforma Render [35]. Embora ofereça serviços pagos, a opção selecionada foi a gratuita, que, apesar de limitada, atende aos requisitos deste trabalho. Similar ao Vercel, o Render integra-se com o GitHub [11], permitindo a entrega contínua e publicação automática de novas versões mediante novos *commits* na *branch* principal. A ferramenta está atualmente acessível online <sup>2</sup>.

Os repositórios dos projetos são de código aberto, alinhados com os princípios do software livre [7]. O código-fonte do *frontend* pode ser acessado neste repositório no GitHub<sup>3</sup>. Já o código-fonte do *backend* está disponível neste outro repositório no GitHub<sup>4</sup>.

A ferramenta criada neste trabalho é liberada sob a licença MIT [7]. Esta licença autoriza qualquer pessoa a usar, copiar, modificar e distribuir o software, desde que a atribuição original seja preservada. A licença MIT, conhecida por sua flexibilidade, é uma das mais populares entre as licenças de código aberto, promovendo a colaboração e o intercâmbio de melhorias entre os desenvolvedores.

## 6 AVALIAÇÃO

O checklist de Pereira *et al.* [31] foi submetido a uma avaliação em um renomado instituto privado de inovação tecnológica. Entramos em contato com o instituto e apresentamos essa nova versão, obtendo a participação de um desenvolvedor e pesquisador, que também fez parte da avaliação da primeira versão do checklist. Além disso, contamos com a valiosa contribuição de três desenvolvedores

<sup>2</sup><https://lcpd-checklist.vercel.app/>

<sup>3</sup><https://github.com/Matheusfrej/LGPD-checklist-frontend>

<sup>4</sup><https://github.com/Matheusfrej/LGPD-checklist-api>

de um núcleo de inovação em tecnologia da informação, comunicação e automação, bem como a avaliação de um desenvolvedor de uma empresa privada. Dessa forma, cinco profissionais experientes no desenvolvimento de sistemas participaram dessa avaliação.

O processo de avaliação consistiu em duas etapas principais: testes práticos da ferramenta e preenchimento de um formulário de feedback. A ferramenta foi testada pelos participantes durante um período específico, após o qual eles responderam a um formulário dividido em três categorias: facilidade de uso, utilidade do uso e intenção de uso. Em cada uma dessas categorias, existem afirmações, em que o entrevistado respondeu o seu grau de concordância com elas, na escala Likert [20]. No final desse formulário, existem campos de resposta aberta para o entrevistado deixar comentários e sugestões adicionais sobre a ferramenta. O método utilizado para a avaliação combinou abordagens quantitativas e qualitativas. Enquanto os comentários se enquadram como feedbacks qualitativos da ferramenta, as afirmações respondidas com o grau de concordância são consideradas quantitativas, a fim de identificar padrões e tendências nas percepções dos participantes.

Com base na avaliação da ferramenta, três participantes concordaram totalmente com sua facilidade de uso, enquanto um concordou parcialmente e outro discordou parcialmente. Em relação à intuição do sistema, três participantes concordaram totalmente, um concordou parcialmente e outro discordou parcialmente. Os participantes identificaram áreas de melhoria para tornar a ferramenta mais acessível e intuitiva, incluindo a sugestão de um participante para tornar o vocabulário da LGPD acessível durante todo o processo de preenchimento, em vez de apenas no início. Isso sugere que a dificuldade encontrada pode estar relacionada à compreensão dos termos legais, em vez da usabilidade direta do sistema.

Além disso, os desenvolvedores apontaram algumas melhorias para a ferramenta, incluindo a utilização de mais gráficos ou outros recursos visuais para tornar a apresentação mais intuitiva e informativa. Aconselharam também destacar campos importantes não preenchidos, como comentários ou seleção de severidade, para facilitar a identificação de áreas que precisam de atenção. Por fim, comentaram que o preenchimento se torna cansativo e, para mitigar essa dificuldade, sugeriram a adição de paginação dentro do checklist, a fim de evitar a necessidade de rolar a tela durante o preenchimento.

Na avaliação da utilidade do Sistema LGPD, observou-se que todos os participantes concordaram que ele permite realizar a inspeção do sistema de forma mais rápida. Em relação à melhoria da produtividade ao identificar defeitos ou problemas no sistema, três concordaram totalmente e dois concordaram parcialmente. Quanto à percepção de utilidade para encontrar defeitos ou problemas no sistema, dois concordaram totalmente, um concordou parcialmente e um discordou parcialmente. Esses resultados indicam uma percepção geral positiva em relação à utilidade do Sistema LGPD para melhorar a inspeção e identificação de problemas no sistema, embora haja ressalvas expressas por alguns participantes.

Com base na intenção de uso do Sistema LGPD, observou-se que dois participantes concordaram totalmente e três concordaram parcialmente que pretendem utilizar a ferramenta para encontrar defeitos ou problemas em sistemas futuramente. Quanto à recomendação do Sistema LGPD para outras empresas, três participantes concordaram totalmente e dois concordaram parcialmente. Esses

resultados sugerem um interesse significativo em utilizar o Sistema LGPD como uma ferramenta para inspeção e identificação de problemas em sistemas, com a maioria dos participantes manifestando intenção de uso e disposição para recomendar a ferramenta para outras empresas.

## 7 CONSIDERAÇÕES FINAIS

Este trabalho apresentou o desenvolvimento de uma ferramenta para facilitar a conformidade das empresas com os requisitos da LGPD. A partir da necessidade de otimizar um checklist em Excel, buscamos oferecer uma solução mais eficiente e acessível para lidar com os desafios regulatórios impostos pela legislação de proteção de dados.

Embora tenhamos feito progressos significativos, reconhecemos que há áreas que podem ser melhoradas. Por exemplo, identificamos que a interface do usuário e a linguagem legal utilizada podem ser aprimoradas para facilitar a compreensão e utilização da ferramenta pelos usuários finais. Além disso, o número limitado de avaliadores pode restringir a generalização dos resultados.

Como próximos passos, planejamos realizar avaliações com um grupo mais amplo de avaliadores, visando obter uma perspectiva mais abrangente sobre a eficácia e usabilidade da ferramenta. Também pretendemos implementar uma nova versão, levando em consideração os feedbacks dos profissionais que participaram desta avaliação. Além disso, planejamos expandir a aplicabilidade da ferramenta para diversas outras áreas de especialização em sistemas, além de IoT.

Ao final, acreditamos que esta ferramenta representa um avanço significativo na simplificação e melhoria dos processos de conformidade com a LGPD, fornecendo às empresas uma solução prática e eficaz para enfrentar os desafios regulatórios em constante evolução.

## DISPONIBILIDADE DOS ARTEFATOS

Os artefatos desenvolvidos durante este trabalho estão disponíveis abertamente em: <https://zenodo.org/records/11397734>

## AGRADECIMENTOS

Este trabalho é parcialmente financiado pelo INES 2.0 ([www.ines.org.br](http://www.ines.org.br)), bolsa CNPq 465614/2014-0, bolsa FACEPE APQ-0399-1.03/17 e APQ/0388-1.03/14, bolsa CAPES 88887.136410/2017- 00. Sérgio Soares é parcialmente apoiado pela bolsa CNPq 306000/2022-9.

## REFERENCES

- [1] Alessandro Acquisti, Laura Brandimarte, and George Loewenstein. 2015. Privacy and human behavior in the age of information. *Science* 347, 6221 (2015), 509–514.
- [2] Stenly Ibrahim Adam, Jimmy H Moedjahedy, and Jeremiah Maramis. 2020. RESTful Web Service Implementation on Unklab Information System Using JSON Web Token (JWT). In *2020 2nd International Conference on Cybernetics and Intelligent System (ICORIS)*. IEEE, Makassar, Indonesia, 1–6. <https://doi.org/10.1109/ICORIS50180.2020.9320801>
- [3] Emerson O Batista. 2017. *Sistemas de informação*. Saraiva Educação SA, Sao Paulo, Brazil.
- [4] D. Baumer, W. Bischofberger, H. Lichter, and H. Zullighoven. 1996. User interface prototyping-concepts, tools, and experience. In *Proceedings of IEEE 18th International Conference on Software Engineering*. IEEE, Berlin, Germany, 532–541. <https://doi.org/10.1109/ICSE.1996.493447>
- [5] Chandradeep Bhatt, Divyanshu Tiwari, Deepika Dua, Shefali Gupta, and Teekam Singh. 2024. Elevating Online Retail: An In-Depth Look at the Implementation of React JS in Advanced E-commerce. In *2024 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE)*. IEEE, New York, NY, USA, 1–4. <https://doi.org/10.1109/IITCEE59897.2024.10467991>

- [6] Brasil. 2018. Lei Geral de Proteção de Dados Pessoais (LGPD). Disponível em: [https://www.planalto.gov.br/ccivil\\_03/\\_ato2015-2018/2018/lei/113709.htm](https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/113709.htm). [https://www.planalto.gov.br/ccivil\\_03/\\_ato2015-2018/2018/lei/113709.htm](https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/113709.htm) Acesso em: 24 de abril de 2024.
- [7] Augusto Campos. 2006. O que é software livre. Disponível em: <http://br-linux.org/linux/faq-softwarelivre>. Acesso em: 15 de maio de 2024.
- [8] Orlando Amaral Cejas, Muhammad Ilyas Azeem, Sallam Abualhaija, and Lionel C. Briand. 2023. NLP-Based Automated Compliance Checking of Data Processing Agreements Against GDPR. *IEEE Transactions on Software Engineering* 49, 9 (2023), 4282–4303. <https://doi.org/10.1109/TSE.2023.3288901>
- [9] Diego André Cerqueira, Rafael Maiani de Mello, and Guilherme Horta Travassos. 2023. Experimental Evaluation of a Checklist-Based Inspection Technique to Verify the Compliance of Software Systems with the Brazilian General Data Protection Law. arXiv:2308.14874 [cs.SE]
- [10] Edgar F Codd. 1990. *The relational model for database management: version 2*. Addison-Wesley Longman Publishing Co., Inc., MA, United States.
- [11] Valerio Cosentino, Javier L Cánovas Izquierdo, and Jordi Cabot. 2017. A systematic mapping study of software development with GitHub. *Ieee access* 5 (2017), 7173–7192.
- [12] Lorrie Faith Cranor. 1999. Internet privacy. *Commun. ACM* 42, 2 (feb 1999), 28–38. <https://doi.org/10.1145/293411.293440>
- [13] Evandro Thalles Vale de Castro, Geovana RS Silva, and Edna Dias Canedo. 2022. Ensuring privacy in the application of the Brazilian general data protection law (LGPD). In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*. ACM, Salamanca, Spain, 1228–1235.
- [14] DevMedia. 2024. ORM (Object-Relational Mapper). Disponível em: <https://www.devmedia.com.br/orm-object-relational-mapper/19056>. Acesso em: 10 de maio de 2024.
- [15] Anirudh Ekambaranathan, Jun Zhao, and George Chalhoub. 2023. Navigating the Data Avalanche: Towards Supporting Developers in Developing Privacy-Friendly Children's Apps. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 7, 2 (2023), 1–24.
- [16] European Parliament and Council of the European Union. 2016. Regulation (EU) 2016/679 of the European Parliament and the Council (General Data Protection Regulation). Disponível em: <https://gdpr-info.eu/>. <https://gdpr-info.eu/> Acesso em: 24 de abril de 2024.
- [17] Fang He. 2023. Implementation of Secure Login and Access Methods for Web Frontend. In *2023 International Conference on Applied Intelligence and Sustainable Computing (ICAISC)*. IEEE, IEEE, New York, NY, USA, 1–5.
- [18] Martti Jeenicke, Wolf-Gideon Bleek, and Ralf Klischewski. 2003. Revealing Web User Requirements through e-Prototyping. In *SEKE*. Citeseer, Citeseer, San Francisco, USA, 9–16.
- [19] Michael Jones, John Bradley, and Nat Sakimura. 2015. *Json web token (jwt)*. Technical Report. Internet Engineering Task Force (IETF).
- [20] A. Joshi, S. Kale, S. Chandel, and D. K. Pal. 2015. Likert Scale: Explored and Explained. *Current Journal of Applied Science and Technology* 7, 4 (2015), 396–403. <https://doi.org/10.9734/BJAST/2015/14975> Acesso em: 26 de abril de 2024.
- [21] Altino Alves Júnior, Leticia de Souza Meireles, Lucas Alves Rossi Figueira, Vítor Marcondes Moraes Carmo, Humberto T Marques-Neto, and Laerte Xavier. 2022. Entendendo o engajamento das comunidades front-end e back-end nos repositórios do GitHub. In *Anais do X Workshop de Visualização, Evolução e Manutenção de Software*. SBC, SBC, Virtual Event, 26–30.
- [22] Gursheen Kaur and Raj Gaurang Tiwari. 2023. Comparison and Analysis of Popular Frontend Frameworks and Libraries: An Evaluation of Parameters for Frontend Web Development. In *2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC)*. IEEE, IEEE, New York, NY, USA, 1067–1073.
- [23] R.A. Kemmerer. 2003. Cybersecurity. In *25th International Conference on Software Engineering, 2003. Proceedings*. IEEE Computer Society, Portland, OR, USA, 705–715. <https://doi.org/10.1109/ICSE.2003.1201257>
- [24] Pedro Machado, Jéssyka Vilela, Mariana Peixoto, and Carla Silva. 2023. A systematic study on the impact of GDPR compliance on Organizations. In *Proceedings of the XIX Brazilian Symposium on Information Systems*. SBC, Maceió, Brazil, 435–442.
- [25] Rodrigo Machado, Diego Kreutz, Giulliano Paz, and Gustavo Rodrigues. 2019. Vazamentos de Dados: Histórico, Impacto Socioeconômico e as Novas Leis de Proteção de Dados. In *Anais da XVII Escola Regional de Redes de Computadores (Alegrete)*. SBC, Porto Alegre, RS, Brasil, 154–159. <https://doi.org/10.5753/errc.2019.9230>
- [26] João Mendes, Davi Viana, and Luis Rivero. 2021. Developing an inspection checklist for the adequacy assessment of software systems to quality attributes of the brazilian general data protection law: An initial proposal. In *Proceedings of the XXXV Brazilian Symposium on Software Engineering*. ACM, Porto Alegre, RS, Brasil, 263–268.
- [27] João Pedro Marques Mendes. 2022. *LGPD-CHECK: Um Checklist para Avaliação da Aderência de Sistemas Computacionais à Lei Geral de Proteção de Dados Brasileira*. Master's thesis. Universidade Federal do Maranhão.
- [28] Microsoft. 2012. TypeScript. Disponível em: <https://www.typescriptlang.org/>. Acesso em: 09 de maio de 2024.
- [29] Cornelia Mihaela Novac, Ovidiu Constantin Novac, Raluca Marina Sferle, Mircea Ioan Gordian, Gyöngyi Bujdosó, and Camelia Maria Dindelegan. 2021. Comparative study of some applications made in the Vue.js and React.js frameworks. In *2021 16th International Conference on Engineering of Modern Electric Systems (EMES)*. IEEE, Oradea, Romania, 1–4. <https://doi.org/10.1109/EMES52337.2021.9484149>
- [30] Fabrício Peloso Piurcosky, Marcelo Aparecido Costa, Rodrigo Franklin Frogeri, and Cristina Lelis Leal Calegario. 2019. A lei geral de proteção de dados pessoais em empresas brasileiras: uma análise de múltiplos casos. *Suma de negocios* 10, 23 (2019), 89–99.
- [31] Ivonildo Pereira, João Mendes, Davi Viana, Luis Rivero, Waldemar Ferreira, and Sergio Soares. 2022. Extending an LGPD Compliance Inspection Checklist to Assess IoT Solutions: An Initial Proposal. In *Trilha da Indústria - Congresso Brasileiro de Software: Teoria e Prática (CBSOFT)*, 13. SBC, Uberlândia, Brazil, 28–31. [https://doi.org/10.5753/cbsoft\\_estendido.2022.226679](https://doi.org/10.5753/cbsoft_estendido.2022.226679)
- [32] José Maurício dos Santos Pinheiro. 2017. Ameaças e Ataques aos Sistemas de Informação: Prevenir e Antecipar. *Cadernos UniFOA* 3, 5 (mar. 2017), 11–21. <https://doi.org/10.47385/cadunifoa.v3.n5.885>
- [33] Patricia Peck Pinheiro. 2020. *Proteção de dados pessoais: Comentários à lei n. 13.709/2018-Igpd*. Saraiva Educação SA, Brasil.
- [34] Prisma Labs, Inc. 2024. Prisma. Disponível em: <https://www.prisma.io/>. Acesso em: 10 de maio de 2024.
- [35] Render, Inc. Accessed 2024. Render - The Easiest Cloud For All Your Apps And Websites. Disponível em: <https://render.com/>. <https://render.com/> Acesso em: 15 de maio de 2024.
- [36] Cedric Ryngaert and Mistale Taylor. 2020. The GDPR as Global Data Protection Regulation? *AJIL Unbound* 114 (2020), 5–9. <https://doi.org/10.1017/aju.2019.80>
- [37] Pattaraporn Sangaroonsilp, Hoa Khanh Dam, Morakot Choetkiertikul, Chaiyong Ragkhitwetsagul, and Aditya Ghose. 2023. A taxonomy for mining and classifying privacy requirements in issue reports. *Information and Software Technology* 157 (2023), 107162.
- [38] Joshua D Scarsbrook, Mark Utting, and Ryan KL Ko. 2023. TypeScript's Evolution: An Analysis of Feature Adoption Over Time. In *2023 IEEE/ACM 20th International Conference on Mining Software Repositories (MSR)*. IEEE, IEEE, Melbourne, Australia, 109–114.
- [39] Mojtaba Shahin, Muhammad Ali Babar, and Liming Zhu. 2017. Continuous integration, delivery and deployment: a systematic review on approaches, tools, challenges and practices. *IEEE access* 5 (2017), 3909–3943.
- [40] Jefferson Silva, Newton Calegari, and Eduardo Gomes. 2019. After Brazil's General Data Protection Law: Authorization in Decentralized Web Applications. In *Companion Proceedings of the 2019 World Wide Web Conference*. ACM, New York, NY, USA, 819–822.
- [41] Sean Sirur, Jason RC Nurse, and Helena Webb. 2018. Are we there yet? Understanding the challenges faced in complying with the General Data Protection Regulation (GDPR). In *Proceedings of the 2nd International Workshop on Multimedia Privacy and Security*. ACM, Toronto, Canada, 88–95.
- [42] Vercel, Inc. Accessed 2024. Vercel - Develop. Preview. Ship. Disponível em: <https://vercel.com/>. <https://vercel.com/> Acesso em: 15 de maio de 2024.
- [43] Rishi Vyas. 2022. Comparative Analysis on Front-End Frameworks for Web Applications. *International Journal for Research in Applied Science and Engineering Technology* 10, 7 (2022), 298–307.