

Leticia Cohen Ferrari, Michel Trindade Brito e Victor Hoefling Padula

**SISTEMA DE APOIO E GERENCIAMENTO  
DE ONGS: INSTÂNCIA DE RESGATE E  
ADOÇÃO DE ANIMAIS**

São Paulo, SP

2024

Leticia Cohen Ferrari, Michel Trindade Brito e Victor Hoefling Padula

# **SISTEMA DE APOIO E GERENCIAMENTO DE ONGS: INSTÂNCIA DE RESGATE E ADOÇÃO DE ANIMAIS**

Trabalho de conclusão de curso apresentado ao Departamento de Engenharia de Computação e Sistemas Digitais da Escola Politécnica da Universidade de São Paulo para obtenção do Título de Engenheiro.

Universidade de São Paulo – USP

Escola Politécnica

Departamento de Engenharia de Computação e Sistemas Digitais (PCS)

Orientador: Prof. Dr. Jorge Luis Risco Becerra

São Paulo, SP

2024

# Resumo

O presente trabalho tem como objetivo desenvolver uma plataforma de gestão para Organizações Não Governamentais (ONGs) com foco na otimização de suas operações. Tomando a ONG Anjos na Terra como referência, uma instância do sistema será projetado para permitir a adição de novas funcionalidades e expansões que atendam a demandas específicas do setor, visando sua aplicação em outras ONGs com finalidades semelhantes, isto é, de resgate e adoção de animais. Espera-se que a expansão do sistema possa ser conduzida por alunos da Escola Politécnica da USP, como parte das Atividades Extensionistas da nova grade curricular, promovendo o envolvimento acadêmico na melhoria contínua da plataforma. O projeto também busca reforçar a conexão entre a universidade e a sociedade, incentivando o impacto social positivo ao apoiar ONGs que desempenham papel fundamental no cuidado e na proteção de animais, promovendo a responsabilidade social e ambiental.

**Palavras-chave:** ONG. plataforma de gerenciamento. universidade e sociedade. transformação digital. terceiro setor.

# Lista de ilustrações

Figura 1 – Iteração sobre duas atividades de resolução de problemas . . . . .	13
Figura 2 – Objetivos de Desenvolvimento Sustentável . . . . .	16
Figura 3 – Alinhamento entre objetivos de negócio, tecnológicos e operacionais . .	18
Figura 4 – Sprints do primeiro quadrimestre de 2024 . . . . .	20
Figura 5 – Sprints do terceiro quadrimestre de 2024 . . . . .	21
Figura 6 – Controle dos Animais da ONG . . . . .	22
Figura 7 – À direita, página exibida ao entrar no QR Code gerado. . . . .	23
Figura 8 – Modelo Organizacional da ONG Anjos na Terra . . . . .	24
Figura 9 – Diagrama BPMN do Processo de Resgate do Animal . . . . .	26
Figura 10 – Diagrama BPMN do Processo de Adoção do Animal . . . . .	26
Figura 11 – Diagrama BPMN do Processo de Gestão de Voluntários e Captação de Recursos . . . . .	27
Figura 12 – Diagrama BPMN do Processo de Administração da ONG . . . . .	27
Figura 13 – Diagrama BPMN do Processo de Criação de Eventos . . . . .	28
Figura 14 – Arquitetura de Plataforma Digital de Referência . . . . .	31
Figura 15 – Arquitetura de Plataforma Digital Projetada . . . . .	32
Figura 16 – Camadas de Arquitetura do Sistema . . . . .	36
Figura 17 – Diagrama de funcionalidades do sistema e usuários . . . . .	36
Figura 18 – Camada de Engenharia da Instância em Resgate e Adoção de Pets . .	37
Figura 19 – Camada de Computação em Microserviços . . . . .	41
Figura 20 – Infraestrutura da Implementação . . . . .	42
Figura 21 – Entidades dos Bancos de dados na Camada de Informação . . . . .	43
Figura 22 – Camada de Interface em Design Atômico . . . . .	44
Figura 23 – Prototipação da Camada de Interface desenvolvido no Figma . . . . .	45
Figura 24 – Repositório de Código Fonte para Frontend . . . . .	51
Figura 25 – Monitoramento do Ambiente de Produção em Nuvem . . . . .	52
Figura 26 – Página Inicial do Sistema . . . . .	53
Figura 27 – Módulo de Administração de Animais - Anjos Na Terra - Versão Desktop e Mobile . . . . .	53
Figura 28 – Navegação entre Módulos - Anjos Na Terra . . . . .	54
Figura 29 – Módulo de Administração Financeira - Anjos Na Terra - Versão Desktop e Mobile . . . . .	54
Figura 30 – Visita à ONG Anjos Na Terra . . . . .	55
Figura 31 – Animal Cadastrado no Sistema - Visão administrativa e visão pública .	56
Figura 32 – Animal Cadastrado com coleira de Identificação . . . . .	56
Figura 33 – Formulário de Cadastro do Animal . . . . .	57

Figura 34 – Formulário de Adoção . . . . . 57

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>7</b>
<b>1.1</b>	<b>Motivação</b>	<b>7</b>
1.1.1	ONGs no Brasil	7
1.1.2	Escopo: Operações da ONG Anjos na Terra	7
<b>1.2</b>	<b>Objetivos</b>	<b>8</b>
<b>1.3</b>	<b>Justificativa</b>	<b>10</b>
1.3.1	Artigo 1: Impactos da pandemia da COVID-19 no processo de transformação digital das ONGs Brasileiras	10
1.3.2	Artigo 2: Como a tecnologia pode ajudar as ONGs	11
<b>1.4</b>	<b>Ideação: <i>Design Science</i></b>	<b>12</b>
<b>1.5</b>	<b>Organização do Trabalho</b>	<b>13</b>
<b>2</b>	<b>ASPECTOS CONCEITUAIS</b>	<b>14</b>
<b>2.1</b>	<b>Modelo de Negócio Social</b>	<b>14</b>
2.1.1	O Terceiro Setor	14
2.1.2	Como as ONG's Sobrevivem	15
<b>2.2</b>	<b>Metas de Desenvolvimento Sustentável</b>	<b>15</b>
<b>2.3</b>	<b>Transformação Digital</b>	<b>16</b>
<b>2.4</b>	<b>Plataforma Digital</b>	<b>17</b>
<b>3</b>	<b>ESPECIFICAÇÃO E DESENVOLVIMENTO</b>	<b>19</b>
<b>3.1</b>	<b>Metodologia Ágil e Planejamento do Projeto</b>	<b>19</b>
3.1.1	<i>Sprints</i>	19
<b>3.2</b>	<b>Análise de Requisitos</b>	<b>21</b>
3.2.1	Contexto, Atores e <i>Stakeholders</i>	21
3.2.2	Análise do Modelo de Negócio	25
3.2.3	Definição de Requisitos na instância de resgate e adoção de animais	28
3.2.3.1	Requisitos Funcionais	28
3.2.3.2	Requisitos Não Funcionais	30
<b>3.3</b>	<b>Arquitetura da Plataforma Digital de Sistemas de Administração para ONGs</b>	<b>30</b>
3.3.1	Plataforma Orientada a Módulos e Integração	32
3.3.2	Funcionalidades de Apresentação e Experiência do Usuário	32
3.3.3	Gestão de Conteúdo e Campanhas	33
3.3.4	Automação e Recomendações Baseadas em IA	33
3.3.5	Segurança e Conformidade	34

3.3.6	Suporte a Escalabilidade e Alta Disponibilidade . . . . .	34
3.3.7	Análise e Relatórios . . . . .	34
3.3.8	Sistemas de E-commerce e Marketplace . . . . .	35
3.3.9	Capacidades Colaborativas . . . . .	35
<b>3.4</b>	<b>Arquitetura em Camadas da Instância de Resgate e Adoção de Pets</b>	<b>35</b>
3.4.1	Camada de Negócios . . . . .	36
3.4.2	Camada de Engenharia . . . . .	37
3.4.3	Camada de Computação . . . . .	39
3.4.4	Camada de Informação . . . . .	42
3.4.5	Camada de Interface . . . . .	44
<b>4</b>	<b>IMPLEMENTAÇÃO . . . . .</b>	<b>46</b>
<b>4.1</b>	<b>Tecnologias Utilizadas . . . . .</b>	<b>46</b>
4.1.1	Microserviços . . . . .	46
4.1.2	Design Patterns . . . . .	46
4.1.3	Arquitetura MVC . . . . .	47
4.1.4	Ecosistema Spring . . . . .	47
4.1.5	JWT Token . . . . .	48
4.1.6	Ecosistema React . . . . .	48
4.1.7	Framework NextJS . . . . .	49
4.1.8	Tailwind CSS . . . . .	49
4.1.9	Conteirização: Docker . . . . .	50
4.1.10	Integração e Desenvolvimento Contínuos (CI/CD) . . . . .	50
<b>4.2</b>	<b>Desenvolvimento e Operação . . . . .</b>	<b>51</b>
4.2.1	Ambientes de Desenvolvimento, Testes e Produção . . . . .	51
4.2.2	Operação . . . . .	52
<b>4.3</b>	<b>Testes de Integração . . . . .</b>	<b>54</b>
<b>5</b>	<b>CONSIDERAÇÕES FINAIS . . . . .</b>	<b>58</b>
<b>5.1</b>	<b>Conclusões do Projeto de Formatura . . . . .</b>	<b>58</b>
<b>5.2</b>	<b>Contribuições . . . . .</b>	<b>59</b>
<b>5.3</b>	<b>Perspectivas de Continuidade . . . . .</b>	<b>59</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>60</b>

# 1 Introdução

Nesta seção será abordada a motivação do projeto (contexto e escopo), os objetivos, as justificativas e a organização do trabalho.

## 1.1 Motivação

### 1.1.1 ONGs no Brasil

Segundo o Instituto de Pesquisa Econômica Aplicada (IPEA, 2024), existem hoje no Brasil, mais de 815 mil Organizações da Sociedade Civil (OSCs), ou Organizações Não Governamentais (ONGs), como são conhecidas popularmente. Ainda segundo o mesmo instituto, o tempo médio de atividade dessas organizações é de 17,55 anos (IPEA, 2020), fato que decorre, principalmente, da dificuldade que essas organizações apresentam para realizar atividades de gestão (IPEA, 2020; GIFE, 2024).

Dentro das principais dificuldades de gestão das ONGs, encontram-se questões como: gestão financeira (levantamento de recursos, transparência e prestação de contas), gestão de voluntários (recrutamento, retenção e organização), temas jurídicos e demonstração de impacto (ECONOMATO, 2024).

Entre as diversas ONGs que atuam no Brasil, destacam-se aqui aquelas que se dedicam ao resgate, cuidado e doação de animais. Essas organizações desempenham um papel crucial na proteção e reabilitação de animais vítimas de abandono, maus-tratos e outras formas de violência, além de promover o bem-estar animal em geral. Segundo um levantamento realizado em 2022, pelo Instituto Pet Brasil, existiam 400 ONGs que atuavam em prol desta causa até o momento em questão, tutelando mais de 184 mil animais (IPB, 2024).

Essas instituições enfrentam uma série de obstáculos distintos que vão desde aqueles mencionados previamente nesta seção até a logística de operações diárias, que envolve o cadastro e controle dos animais, entre outros. Tais ONGs costumam operar em um contexto onde a demanda por assistência é constante e muitas vezes imprevisível, enquanto os recursos disponíveis são limitados e variáveis.

### 1.1.2 Escopo: Operações da ONG Anjos na Terra

O problema que esta monografia busca resolver está diretamente ligado à falta de um sistema eficiente e estruturado para gerenciar as operações da ONG Anjos na Terra.

Atualmente, a ONG lida com um volume considerável de dados e processos de forma rudimentar, o que gera ineficiências operacionais.

A ausência de um controle automatizado e integrado para o cadastro de animais, o gerenciamento de feiras de adoção, o controle de voluntários e a gestão financeira acarreta uma sobrecarga significativa, especialmente para a fundadora, Fátima, que centraliza grande parte das atividades.

Além disso, a falta de uma hierarquia clara e de uma definição precisa das funções dos voluntários contribui para a desorganização, o que impacta diretamente a capacidade da ONG de operar de maneira otimizada.

Ademais, o uso de ferramentas inadequadas, como cadernos e, no melhor dos casos, de planilhas do Google, para o controle dos animais, aliada à limitação de recursos e à gestão manual de dados, agrava o problema.

Com isso, a falta de um modelo estruturado de organização e o uso de soluções prontas e improvisadas traz dificuldades na distribuição de informações e tarefas dentro da ONG. Não obstante, a ausência de um controle de dados promove problemas de visibilidade da organização além de diminuir a arrecadação de recursos e parceiros.

Assim, o desafio central que este trabalho propõe resolver é a criação de uma plataforma integrada de gerenciamento que permita automatizar e organizar os processos da ONG, facilitando seu funcionamento e possibilitando sua expansão de maneira mais eficiente. Reconhecendo a complexidade envolvida no desenvolvimento de uma plataforma com técnicas clássicas de engenharia de software, especialmente devido às exigências de distribuição e escalabilidade, este projeto adota uma abordagem orientada à arquitetura em camadas.

Essa abordagem busca garantir flexibilidade, modularidade e robustez, atendendo às demandas operacionais e permitindo a transformação digital da organização de forma sustentável.

## 1.2 Objetivos

O objetivo deste trabalho é desenvolver uma plataforma digital, baseada em uma arquitetura modular e escalável de microserviços, para a gestão de Organizações Não Governamentais (ONGs) que atuam em diversas áreas, com a primeira aplicação voltada para ONGs que resgatam e promovem a adoção de animais. O sistema busca automatizar, centralizar e otimizar processos essenciais dessas organizações, tomando como referência as operações da ONG Anjos na Terra. A proposta inclui a criação de um projeto de extensão extracurricular na Escola Politécnica da USP, reforçando o compromisso social da universidade em devolver à sociedade o investimento feito na formação dos engenheiros.

A complexidade deste produto reside em sua modularidade, escalabilidade e capacidade de integração, características indispensáveis para atender às diferentes necessidades de ONGs de portes variados. A arquitetura em microserviços permite que novos módulos sejam desenvolvidos e integrados gradualmente, sem comprometer a estabilidade do sistema principal. Cada módulo aborda um domínio específico da gestão de ONGs, como cadastro e acompanhamento de animais, gestão financeira, administração de voluntários e controle de eventos, mantendo a coerência de dados entre as áreas e assegurando a flexibilidade necessária para futuras expansões.

Tecnologias escaláveis, como bancos de dados distribuídos e serviços baseados em nuvem, serão consideradas para garantir que a plataforma suporte o aumento no volume de dados e operações das ONGs à medida que elas expandem suas atividades, além de comportar um grande número de organizações.

No primeiro módulo, focado na gestão de animais, será implementado um sistema que controla o ciclo de vida dos resgates, incluindo entrada e saída, dados de saúde (como vacinação) e histórico de adoções. Esse processo apresenta desafios como o gerenciamento de dados sensíveis e a criação de interfaces intuitivas para usuários com diferentes níveis de familiaridade tecnológica. Outro ponto crítico será a implementação de mecanismos para garantir a consistência de dados em tempo real, mesmo em cenários de alta carga transacional.

Além disso, será desenvolvida uma funcionalidade específica para o gerenciamento de eventos de adoção, integrando informações dos animais com o registro de adotantes e a gestão de candidaturas. O sistema buscará equilibrar simplicidade de uso com a robustez necessária para apoiar decisões críticas no processo de adoção, como a análise de perfis de candidatos.

Outro módulo importante abordará a gestão financeira das ONGs, incluindo captação de recursos, prestação de contas e transparência nas operações. Este módulo exige a implementação de soluções que combinem segurança de dados financeiros com acessibilidade para o público interessado, contribuindo para a credibilidade da organização.

Por fim, a gestão de voluntários será estruturada em torno de um sistema capaz de organizar diferentes tipos de contribuição, como trabalho presencial, doações financeiras, materiais e lares temporários. O maior desafio neste módulo será criar uma solução flexível o suficiente para acomodar as especificidades de cada ONG, garantindo ao mesmo tempo uma visão consolidada das operações.

Embora o foco inicial seja na implementação de soluções específicas para a ONG Anjos na Terra, o sistema será projetado para ser escalável e expansível. Futuras integrações poderão incluir a conexão com sistemas municipais para facilitar o compartilhamento de informações e a colaboração entre ONGs em nível local, além da criação de portais para

divulgar dados como animais disponíveis para adoção ou doações necessárias, permitindo maior envolvimento do público.

A plataforma também poderá ser estendida para possuir integração com redes sociais para ampliar o alcance das campanhas e promover a conscientização sobre as causas das ONGs. Além disso, será possível adaptar o sistema para outras áreas de atuação social, como por exemplo o mapeamento de centros de alimentação para moradores de rua ou a assistência a idosos em situação de vulnerabilidade. Dessa forma, o sistema, além de atender às necessidades da ONG Anjos na Terra, poderá se expandir para diversas causas sociais, ampliando o impacto positivo gerado.

Espera-se que essas melhorias possam ser trabalhadas como atividades extensionistas da Escola Politécnica da USP, fortalecendo a conexão entre a universidade e a sociedade, ao mesmo tempo em que alunos têm a oportunidade de aplicar conhecimentos técnicos em projetos que gerem uma contribuição benéfica e significativa para a sociedade.

Sendo assim, este projeto se destaca não apenas pela sua complexidade técnica, mas também pela sua contribuição direta ao fortalecimento das ONGs e pela promoção de um impacto social mais amplo, demonstrando como a engenharia pode servir como um vetor de transformação social.

## 1.3 Justificativa

Uma plataforma de gerenciamento para ONGs pode ser muito eficaz para otimizar a eficiência dessas organizações. Trabalhos anteriores mostram que ONGs muitas vezes enfrentam desafios devido à falta de infraestrutura adequada, o que dificulta a gestão de seus recursos e operações. Uma plataforma integrada ajuda a superar esses obstáculos ao centralizar informações, automatizar tarefas e melhorar a comunicação e colaboração interna e externa, aumentando o seu impacto social e focando os seus esforços nos trabalhos operacionais que trazem mais valor à comunidade.

Estudos sobre transformação digital em ONGs indicam que essas ferramentas tecnológicas aumentam a produtividade, melhoram a gestão de voluntários e doações, e ampliam o impacto social por meio de uma comunicação mais eficaz com o público e possíveis doadores. Além disso, a digitalização facilita a transparência financeira, o que é vital para manter a confiança pública e a eficácia operacional.

### 1.3.1 Artigo 1: Impactos da pandemia da COVID-19 no processo de transformação digital das ONGs Brasileiras

No contexto brasileiro, a importância da transformação digital (TD) nas ONGs tem sido amplamente reconhecida, sobretudo pela sua capacidade de otimizar operações,

ampliar o alcance de campanhas e melhorar o engajamento com a comunidade. Um estudo realizado após a pandemia (MEIRELLES et al., 2023) destacou o aumento significativo no uso de tecnologias digitais por ONGs, impulsionado pela necessidade de adaptação a um cenário de distanciamento social e escassez de recursos físicos. Esse movimento mostrou às próprias organizações a relevância da digitalização como um caminho para garantir sua sustentabilidade, aumentando a eficiência operacional e a transparência.

No entanto, as ONGs enfrentam barreiras significativas em sua jornada para a digitalização. As organizações consultadas no estudo indicam que, apesar de terem adquirido consciência da importância das tecnologias digitais, a baixa disponibilidade de recursos e o risco envolvido são fatores que as impedem de priorizar essa transformação. Uma das ONGs afirmou que há grande resistência à experimentação, especialmente quando envolve custos financeiros, por conta da incerteza nos retornos: "A questão da experimentação é frágil, temos pouca tolerância ao erro e, quando precisamos de dinheiro para gastar com algo que não seja finalidade, há questionamentos" (MEIRELLES et al., 2023). Outro depoimento reforçou essa dificuldade, destacando que "o risco é um tabu... nós vivemos de doação, então o risco é sempre um problema" (MEIRELLES et al., 2023).

Diante desse cenário, o presente trabalho se propõe como uma alternativa viável para mitigar esses desafios. Ao desenvolver uma solução digital gratuita para o gerenciamento das operações das ONGs, visa-se oferecer uma plataforma acessível que elimine as barreiras financeiras e possibilite a implementação de tecnologias de forma segura e sem riscos elevados, promovendo, assim, uma gestão mais eficiente e transparente.

### 1.3.2 Artigo 2: Como a tecnologia pode ajudar as ONGs

O estudo do Portal do Impacto (FRANÇA, 2021) mostra como as ferramentas digitais não apenas otimizam a gestão interna das ONGs, mas também são fundamentais para construir uma imagem de credibilidade entre apoiadores e doadores. Um sistema de gerenciamento eficiente, como a plataforma proposta neste projeto, pode melhorar a transparência e a organização das operações da ONG, o que é crucial para conquistar e manter a confiança de quem está disposto a contribuir. A tecnologia permite que os apoiadores tenham uma visão mais clara e segura de como suas doações estão sendo utilizadas, aumentando a probabilidade de doações recorrentes e contínuas.

Além disso, o estudo destaca que o uso de tecnologias facilita a comunicação e o relacionamento com os doadores. ONGs que utilizam sistemas modernos para gerenciar suas operações e interações com doadores passam uma imagem de profissionalismo e eficiência, enquanto métodos ultrapassados e confusos podem afastar apoiadores (FRANÇA, 2021). Nesse sentido, a plataforma de gerenciamento que será desenvolvida proporcionará uma interface organizada e acessível, onde a transparência financeira e operacional será um pilar central. Isso não apenas poderá aumentar a credibilidade da ONG, mas também permitirá

uma maior velocidade no processamento de informações e na prestação de contas, aspectos essenciais para a sustentabilidade a longo prazo.

Um exemplo prático trazido pelo Portal do Impacto é o caso da Aliança de Misericórdia, que, ao implementar um sistema de gestão empresarial (ERP), conseguiu organizar seu complexo sistema de receitas e despesas, o que resultou em uma prestação de contas mais ágil e segura (FRANÇA, 2021). Essa experiência reforça como ferramentas digitais podem impactar diretamente a capacidade de uma ONG operar de forma eficaz e transparente, algo que a plataforma de gerenciamento de resgate de animais a ser desenvolvida também busca alcançar. Ao adotar essas tecnologias, as ONGs poderão não apenas otimizar sua gestão interna, mas também aumentar a confiança dos seus apoiadores, ampliando seu impacto social.

## 1.4 Ideação: *Design Science*

A metodologia aplicada neste trabalho foi o Design Science (DS), um método de pesquisa utilizado para resolver problemas práticos por meio da criação de artefatos tecnológicos. O DS é amplamente empregado no desenvolvimento de soluções inovadoras no campo da engenharia de software e sistemas de informação. A escolha dessa metodologia foi motivada pela necessidade de criar uma plataforma de gerenciamento digital que atenda às demandas operacionais da ONG Anjos na Terra.

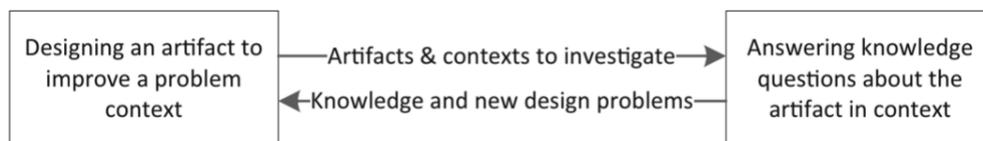
O processo buscou seguir as etapas definidas pelo *Building Digital Experience Platforms - A Guide to Developing Next-Generation Enterprise Applications* (SHIVAKUMAR; SETHI, 2019) e do Design Thinking (SOUZA; FERREIRA; CONTE, 2017). A metodologia envolveu as seguintes fases:

- 1. Identificação do Problema e Definição dos Requisitos:** O primeiro passo foi compreender as necessidades da ONG por meio de entrevistas com uma representante, onde foram levantados os desafios enfrentados, como a gestão de animais, voluntários e o controle financeiro. A partir dessas entrevistas, foi criado um conjunto detalhado de requisitos funcionais e não funcionais para a plataforma.
- 2. Desenvolvimento do Design da Solução:** Após a análise dos requisitos, foi desenvolvido o design inicial da plataforma utilizando a ferramenta Figma para criar protótipos das interfaces. Nessa fase, aplicamos os princípios de design centrado no usuário, buscando simplicidade e eficiência nas interações.
- 3. Prototipagem e Desenvolvimento:** Com o design aprovado, a próxima etapa foi a construção do *frontend* da plataforma, usando as tecnologias *React*, *Typescript*, *Next.js* e *Tailwind*. Paralelamente, o *backend* foi desenvolvido em microserviços,

utilizando *Java Spring*, e integrado ao *frontend*, criando um sistema funcional capaz de suportar as operações atuais da ONG e as futuras expansões.

4. **Validação:** A validação deve ser realizada em três etapas: a apresentação das telas do protótipo no *Figma*, seguida da apresentação do *frontend* funcional e, por fim, a apresentação da integração do sistema completo (*frontend* e *backend*). Cada etapa deve ser revisada e ajustada conforme o *feedback* da ONG, garantindo que o sistema atenda às suas necessidades reais.
5. **Entrega do Artefato:** Após as validações, a plataforma será submetida a testes de aceitação e, uma vez aprovada, será entregue à ONG. Nessa fase, realizaremos testes de integração e testes funcionais para garantir a operação correta do sistema.

Figura 1 – Iteração sobre duas atividades de resolução de problemas



Fonte: (WIERINGA, 2014, p. 5)

Essa abordagem permitiu desenvolver uma solução prática, robusta e adaptada ao contexto da ONG, alinhando-se aos princípios do *Design Science*, que visa não apenas resolver problemas específicos, mas também contribuir com o conhecimento aplicado na criação de artefatos inovadores.

## 1.5 Organização do Trabalho

Inicialmente, buscou-se entender conceitos que serão aplicados no desenvolvimento do trabalho, detalhados no Capítulo 2. Após ter se criado uma base conceitual sólida dos aspectos a serem abordados, o Capítulo 3 detalha o processo de Engenharia de Software: o planejamento da equipe, o levantamento de requisitos e definição da arquitetura. Depois, o Capítulo 4 apresenta o desenvolvimento e detalhes da implementação do sistema. Por fim, o Capítulo 5 traz as conclusões e contribuições da tese, e introduz perspectivas de continuidade do projeto.

## 2 Aspectos Conceituais

Nesta seção serão abordados conceitos essenciais para as definições da arquitetura do projeto, passando pelo estudo do contexto, atores envolvidos e das soluções propostas.

### 2.1 Modelo de Negócio Social

Um modelo de negócio social é um tipo de organização que busca combinar sustentabilidade financeira com a criação de valor social. Diferentemente dos modelos tradicionais, onde o foco está na maximização do lucro, o modelo de negócio social vê o lucro econômico apenas como um meio para garantir sua viabilidade financeira. Ou seja, são desenvolvidos para atender a algum tipo de demanda social e com atuação rentável. (PETRINI; SCHERER; BACK, 2016)

As motivações para a criação de um negócio social, conforme explorado em (MELLO, 2016), incluem o desejo de enfrentar problemas sociais complexos que as políticas públicas e as organizações tradicionais não conseguem resolver de maneira eficaz e sustentável. O cenário de exclusão social, concentração de renda e ineficácia governamental tem motivado indivíduos a desenvolver soluções que integrem impacto social e sustentabilidade financeira. Esse impulso é ainda fortalecido pela crença de que o empreendedorismo social pode servir como alternativa viável para transformar a realidade social, proporcionando uma maneira de inovar no combate às desigualdades e contribuir para uma sociedade mais justa.

Nesta seção será analisado o cenário dos modelos de negócio social no Brasil, englobados no termo "Terceiro Setor", e será feita uma análise do funcionamento dessas organizações.

#### 2.1.1 O Terceiro Setor

Como descrito em (ALBUQUERQUE, 2006), o terceiro setor é composto por organizações sem fins lucrativos que se diferenciam tanto do governo quanto do mercado. Essas instituições, formalmente organizadas e dotadas de uma estrutura interna estável, atuam de forma autônoma, administrando suas próprias atividades sem interferência estatal. São caracterizadas por não distribuírem lucros aos seus proprietários ou administradores, e promovem uma participação cidadã ou voluntária significativa, sendo constituídas livremente por grupos de pessoas. Além disso, desempenham um papel complementar nas ações públicas e privadas, contribuindo para a vida pública com uma visão integradora e enfatizando a importância dos interesses coletivos sobre os individuais. No Brasil, segundo o Instituto de Pesquisa Econômica Aplicada (IPEA, 2023), além do impacto social, o

terceiro setor é responsável por parte significativa da movimentação da economia brasileira, correspondendo a 4,27% do PIB e 5,88% dos empregos gerados direta e indiretamente.

### 2.1.2 Como as ONG's Sobrevivem

O cenário de sobrevivência das organizações não governamentais (ONGs) no Brasil, conforme (CGI, 2022), revela um contexto complexo influenciado pela pandemia de COVID-19. Embora tenha havido avanços em infraestrutura e no uso de tecnologias da informação e comunicação (TIC), as ONGs ainda enfrentam desafios significativos em termos de conectividade e recursos.

Em 2022, 80% das ONGs contavam com áreas administrativas e 68% tinham departamentos de finanças. O financiamento ainda depende fortemente de doações, com 63% das organizações recebendo contribuições de indivíduos, o que representa um aumento em relação a 2016. Contudo, para o mesmo período analisado, houve uma diminuição no acesso a recursos governamentais.

As ONGs mostraram um aumento no uso de celulares e computadores, mas a maioria utilizava dispositivos pessoais em vez de equipamentos fornecidos pelas organizações. As principais atividades na Internet são relacionadas à comunicação e captação de recursos, com um aumento no recebimento de doações online, alcançando 22% em 2022. Entretanto, a presença online é ainda restrita, com a maioria das ONGs sem website próprio, embora haja um crescimento na presença em redes sociais.

Em resumo, as ONGs no Brasil estão em um processo de adaptação e crescimento, mas os desafios financeiros e tecnológicos continuam a ser obstáculos significativos para sua operação e impacto. A situação de funcionamento das organizações sociais, conforme (MELLO, 2016), mostra que muitas enfrentam desafios financeiros e estruturais, com modelos de financiamento tradicionais insuficientes para promover soluções escaláveis. Esse contexto revela a necessidade de novas abordagens que permitam a autossustentabilidade e aumento na captação de recursos.

## 2.2 Metas de Desenvolvimento Sustentável

"Os Objetivos de Desenvolvimento Sustentável são um apelo global à ação para acabar com a pobreza, proteger o meio ambiente e o clima e garantir que as pessoas, em todos os lugares, possam desfrutar de paz e de prosperidade." (ONU, 2024). A Organização das Nações Unidas promove e auxilia países no cumprimento dessas metas e desafios de desenvolvimento até 2030.

Figura 2 – Objetivos de Desenvolvimento Sustentável



Fonte: (ONU, 2024)

Este projeto está alinhado em particular com os objetivos de promover cidades e comunidades sustentáveis (objetivo 11) e proteger a vida terrestre (objetivo 15). Ao desenvolver um sistema de gestão para uma ONG de resgate de animais, busca-se não apenas otimizar os processos internos da organização, mas também contribuir para a melhoria da qualidade de vida dos animais em situação de vulnerabilidade, promovendo o bem-estar animal e a sustentabilidade ambiental.

## 2.3 Transformação Digital

Transformação digital pode ser definida como “um processo de mudança fundamental habilitado por tecnologias digitais, que visa trazer melhorias e inovações radicais para uma entidade (por exemplo, uma organização, uma rede de negócios, uma indústria ou a sociedade) para criar valor para seus stakeholders por meio do aproveitamento estratégico de seus principais recursos e capacidades.” (GONG; RIBIERE, 2021) Esse processo é essencial para a modernização de modelos de negócios, alavancando tecnologias digitais para criar valor e manter a competitividade.

Em estudo publicado pela Harvard Business Review (LAMARRE SHITAL CHHEDA, 2023), destaca-se que as empresas líderes em transformação digital conseguem obter retornos significativamente maiores ao redesenhar processos de ponta a ponta, promovendo a inovação e a eficiência em toda a cadeia de valor. No mesmo estudo, em um exemplo do setor bancário, essas instituições líderes registraram um retorno anual total aos acionistas de 8,1%, contra 4,9% dos retardatários entre 2018 e 2022.

O sucesso das organizações líderes advém de uma abordagem completa e integrada, que vai além da digitalização de pontos de contato. Essas empresas reestruturam seus processos, e estruturam sua organização em modelos. A implementação de ferramentas avançadas, construídas de forma estruturada, junto com uma arquitetura de dados moderna, possibilita uma melhor distribuição da informação e uma automação em escala, tornando os processos da organização mais eficientes.

Assim, a transformação digital bem-sucedida é mais do que simplesmente implementar novas tecnologias; trata-se de uma reestruturação abrangente dos processos e da cultura organizacional. Organizações que adotam essas práticas, impulsionando a inovação e a eficiência, conseguem resultados superiores e uma vantagem competitiva sustentável em um cenário cada vez mais digital.

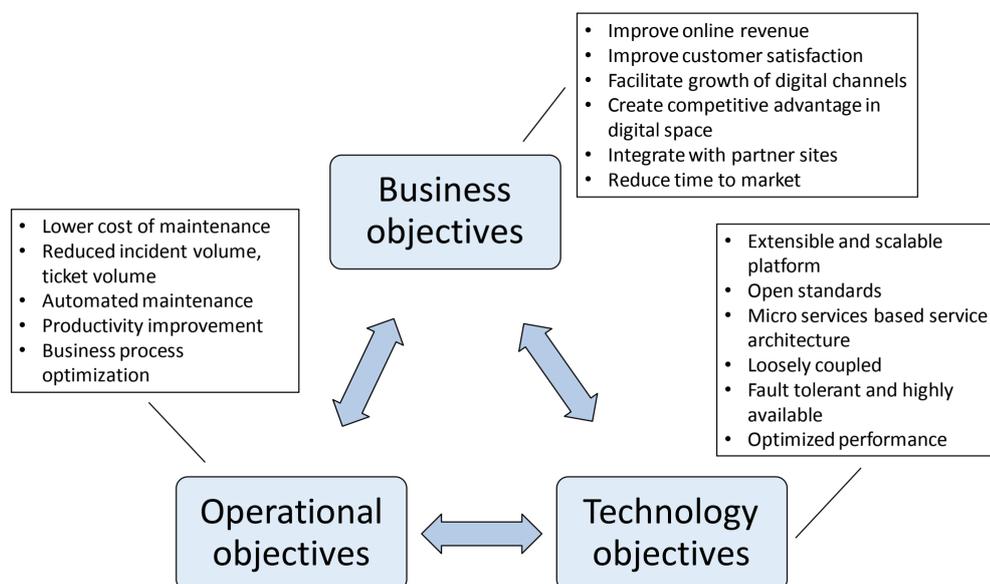
## 2.4 Plataforma Digital

Nesta seção vamos analisar os principais conceitos em torno de uma plataforma digital, partindo do exemplo da plataforma desenvolvida para sistemas bancários em (SHIVAKUMAR; SETHI, 2019). Os autores definem como itens essenciais de uma plataforma digital:

- Visão integrada de todos os dados relevantes;
- Experiência personalizada de entrega da informação baseada em papéis dos atores envolvidos;
- Interatividade e responsividade das interfaces;
- Ferramentas facilitadoras de busca de informação e tomada de decisão;
- Transações seguras e simples de realizar;
- Agilidade na implementação de novas *features*;
- Sistema modular e escalável.

Com isso, conclui-se que os requisitos essenciais para uma plataforma digital de gestão devem garantir a segurança, disponibilidade, interatividade, escalabilidade e otimização dos processos da organização. Dessa maneira, definindo-se objetivos tecnológicos que promovam esses requisitos, torna-se possível alinhar as necessidades de negócio como aumento da captação de recursos e vantagem competitiva, em conjunto com as necessidades operacionais tais quais a otimização de processos e aumento da produtividade.

Figura 3 – Alinhamento entre objetivos de negócio, tecnológicos e operacionais



Fonte: (SHIVAKUMAR; SETHI, 2019, p. 19)

A definição dos requisitos tecnológicos passa pela arquitetura do sistema em camadas, que será discutida no próximo capítulo.

## 3 Especificação e Desenvolvimento

Nesta seção será abordado o processo de desenvolvimento passando pela definição da metodologia de produção, análise de requisitos, e definição da arquitetura.

### 3.1 Metodologia Ágil e Planejamento do Projeto

A metodologia escolhida para o desenvolvimento e implementação do projeto foi a Metodologia Ágil, um método popular na Engenharia de Software. Essa metodologia é uma abordagem de desenvolvimento de software que surgiu em 2001 com a publicação do Manifesto Ágil, criado por um grupo de desenvolvedores e consultores (BECK et al., 2001).

No presente projeto, a Metodologia Ágil foi aplicada no por meio de um planejamento estruturado em *sprints*, permitindo a divisão do trabalho em ciclos iterativos e incrementais. Cada *sprint* foi planejada para abordar uma etapa específica do desenvolvimento, com atividades detalhadas e objetivos claros. Durante as *sprints*, a monografia foi redigida paralelamente.

O processo seguiu os seguintes passos:

1. **Mapeamento e Planejamento das *Sprints*:** O projeto foi dividido em *sprints* com base nas etapas essenciais para sua conclusão. Cada *sprint* tinha um conjunto definido de tarefas a serem realizadas e entregáveis específicos.
2. **Detalhamento das Atividades:** No início de cada *sprint*, as atividades foram detalhadas em termos de escopo e prioridade, garantindo clareza e foco para o desenvolvimento.
3. **Execução Iterativa:** As atividades planejadas para cada *sprint* foram desenvolvidas em ciclos iterativos, permitindo ajustes e refinamentos conforme necessário.

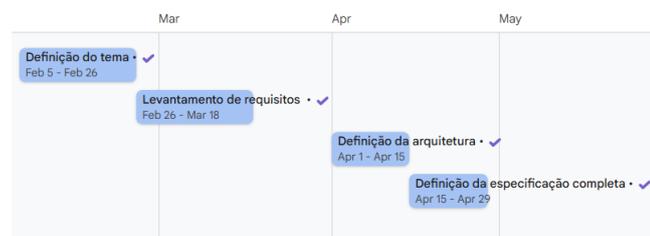
#### 3.1.1 *Sprints*

As sprints planejadas para o projeto foram:

1. **Definição do tema:** Seleção do tema da monografia e alinhamento inicial das ideias principais com orientador.
2. **Levantamento de Requisitos:** Identificação dos requisitos funcionais e não funcionais do sistema.

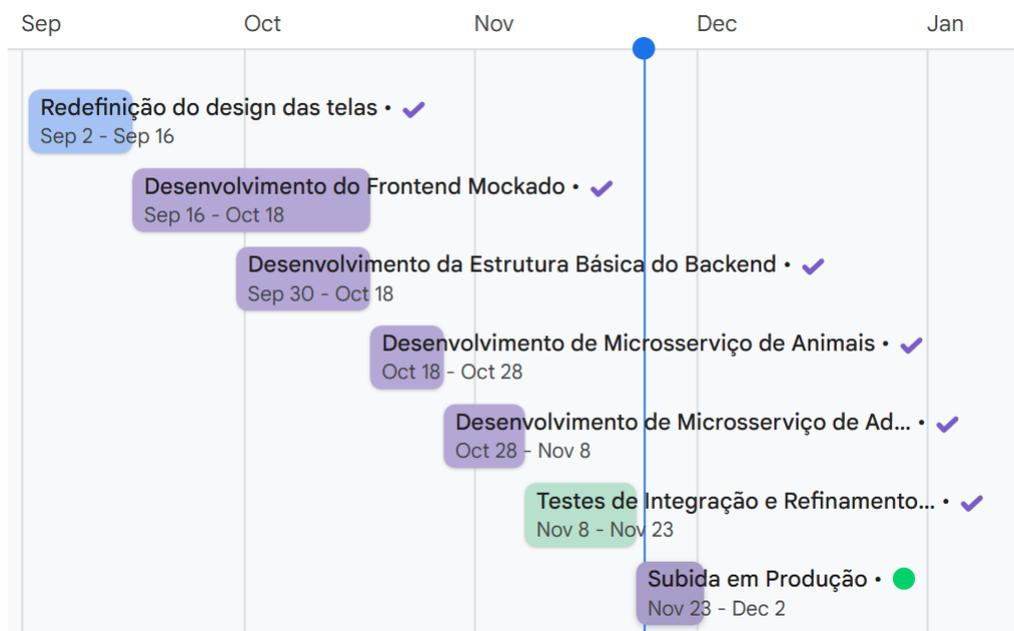
3. **Definição da Arquitetura:** Estruturação inicial da arquitetura do sistema, incluindo a escolha de tecnologias e *frameworks*.
4. **Definição da Especificação Completa:** Elaboração das *features* e design inicial das telas, consolidando a visão do sistema. → **Entregável:** Definição de *features* e design inicial das telas.
5. **Redefinição do Design das Telas:** Refinamento do design de acordo com os *feedbacks* obtidos. → **Entregável:** Novo design de telas.
6. **Desenvolvimento do *Frontend Mockado*:** Implementação inicial do *frontend* com dados simulados para testes visuais. → **Entregável:** *Frontend mockado*.
7. **Desenvolvimento da Estrutura Básica do *Backend*:** Configuração inicial dos microsserviços e integração básica. → **Entregável:** Microsserviços configurados.
8. **Microsserviço de Animais:** Desenvolvimento e integração com o *frontend*.
9. **Microsserviço de Adoção:** Desenvolvimento e integração com o *frontend*.
10. **Testes de Integração e Refinamento da Solução (Local):** Verificação da compatibilidade entre os microsserviços e o *frontend*, corrigindo eventuais problemas identificados.
11. **Subida em Produção:** Publicação da plataforma, tornando-a acessível para uso real.

Figura 4 – Sprints do primeiro quadrimestre de 2024



Fonte: BRITO, COHEN, PADULA 2024

Figura 5 – Sprints do terceiro quadrimestre de 2024



Fonte: BRITO, COHEN, PADULA 2024

## 3.2 Análise de Requisitos

A análise dos requisitos do sistema surge primeiro de um estudo do contexto, seguido da definição das funcionalidades principais.

### 3.2.1 Contexto, Atores e *Stakeholders*

Para identificar os requisitos do sistema que será desenvolvido, é essencial entender o contexto em que a ONG Anjos na Terra opera, assim como os principais atores e *stakeholders* envolvidos. O grupo iniciou a análise entrando em contato diretamente com a ONG para compreender suas necessidades e como a plataforma digital poderia otimizar seus processos. Durante as conversas, foram levantados pontos cruciais relacionados à gestão e operação da ONG, que orientaram as especificações do sistema.

A ONG Anjos na Terra, que trabalha com o resgate e adoção de animais, lida atualmente com um volume considerável de dados, mas de forma bastante rudimentar. Não há um controle estruturado de variáveis importantes para a otimização dos processos internos, como o cadastro de animais, a organização das feiras de adoção, a gestão de voluntários e o controle financeiro. Isso implica que os processos se tornam lentos, sobrecarregados e dependentes de soluções manuais que dificultam a eficiência operacional.

Além disso, também foi mostrado como é feito o controle dos gatos da ONG, que

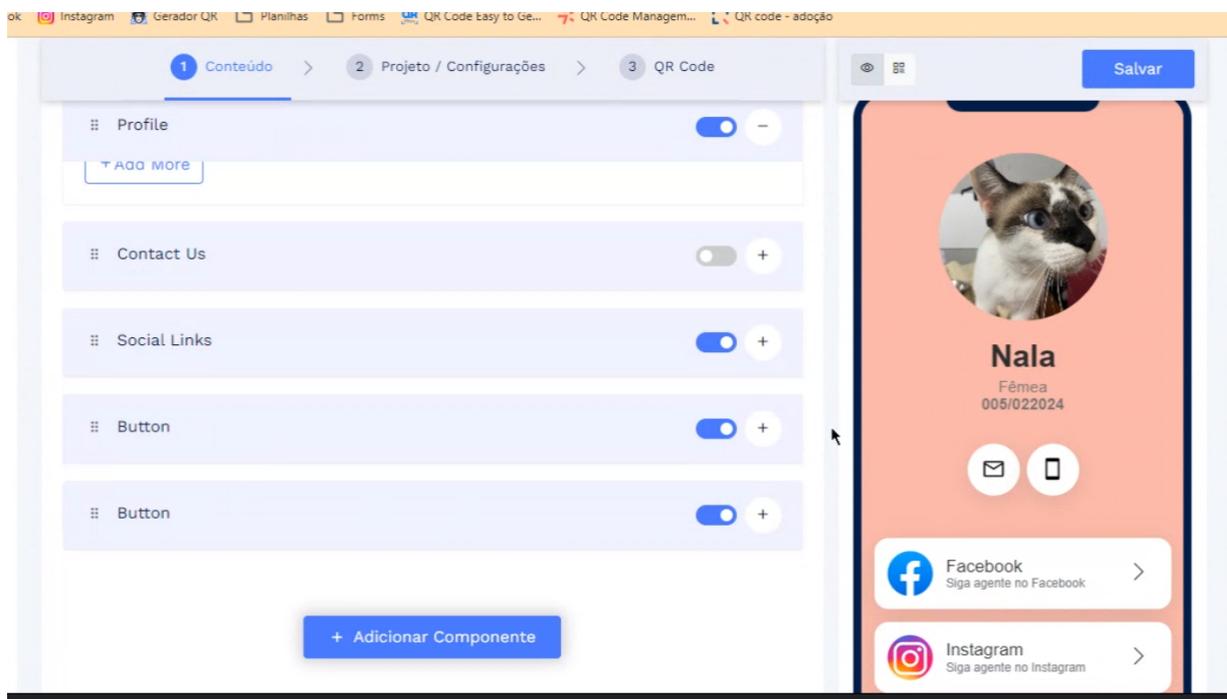
ocorre através de planilhas do Google. Contudo, nem todos os animais estão nelas, apenas os que foram cadastrados numa plataforma que gera QR Codes com as informações do animal. Uma das causas identificadas para o registro parcial do grupo de gatos decorre da falta de organização gerencial e de recursos, uma vez que o processo de cadastro é feito somente por uma pessoa, que utiliza uma plataforma com funcionalidades gratuitas limitadas. Os recursos gerenciais mencionados podem ser observados nas figuras 6 e 7.

Figura 6 – Controle dos Animais da ONG

	A	B	C	D	E	F	G	H	I
1	ID	Nome	Sexo	Imagem	QR Code	17/02/2024	Adoção	Foto da Adoção	Local
3	002/022024	Belinha 002/022024	Fêmea			Adotado	25/02/2024		Cobasi
4	003/022024	Cindy 003/022024	Fêmea			Não vai hoje			
5	004/022024	Frozen 004/022024	Fêmea			Vai hoje			
6	005/022024	Nala 005/022024	Fêmea			Não vai hoje	27/02/2024		
7	006/022024	Bob 006/022024	Macho			Adotado	04/02/2024		Cobasi
8	007/022024	Theo 007/022024	Macho			Não vai hoje			

Fonte: ONG Anjos Na Terra 2024

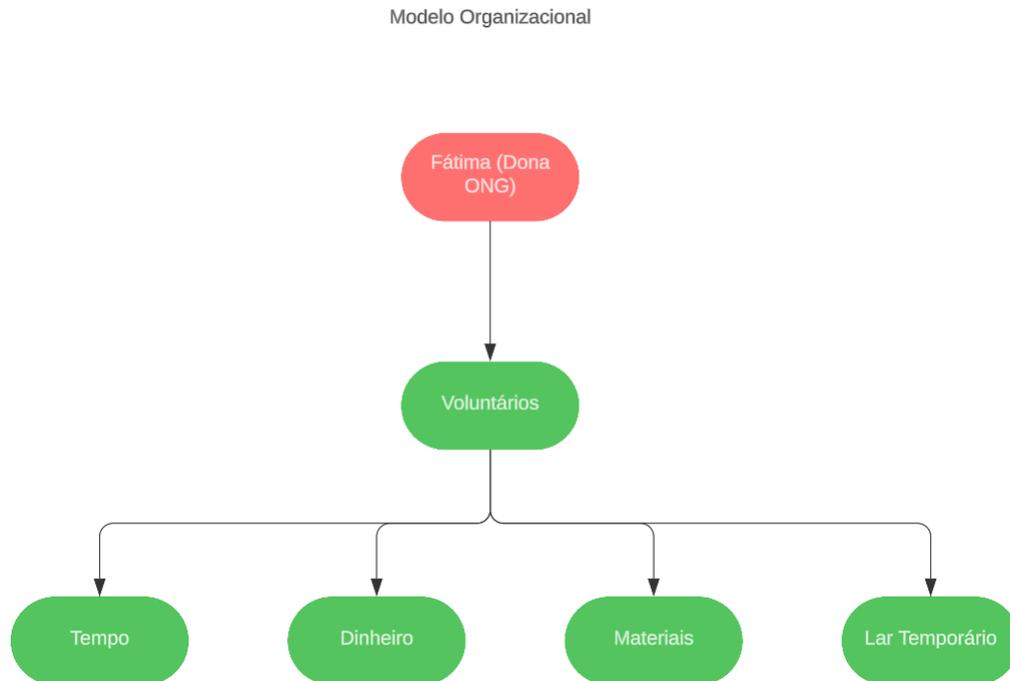
Figura 7 – À direita, página exibida ao entrar no QR Code gerado.



Fonte: ONG Anjos Na Terra 2024

O principal ator que influencia o funcionamento da ONG é a fundadora, Fátima, que gerencia a organização de maneira centralizada. Devido à ausência de uma estrutura hierárquica formal, ela acumula diversas responsabilidades, incluindo o controle de voluntários, a gestão dos processos de adoção e a captação de recursos. Esse modelo de gestão centralizada gera uma sobrecarga significativa de funções, dificultando a eficiência nas operações e limitando a possibilidade de delegar tarefas.

Figura 8 – Modelo Organizacional da ONG Anjos na Terra



Fonte: BRITO, COHEN, PADULA 2024

Outro ator fundamental são os voluntários, que podem ser classificados em quatro grupos: os que dedicam tempo, os que fazem doações financeiras, os que oferecem materiais e aqueles que disponibilizam lares temporários para os animais. Esses grupos, apesar de não serem exclusivos, têm funções distintas e contribuem de maneira significativa para o funcionamento da ONG. No entanto, a falta de uma definição clara sobre as funções de cada grupo e a sobreposição de responsabilidades dificultam a gestão dos recursos humanos e resultam em ineficiências operacionais.

Além de ser uma figura central na operação interna, Fátima é também um *stakeholder* chave, pois suas necessidades de melhoria nos processos administrativos e a busca por maior eficiência serão diretamente atendidas pela presente plataforma. O sistema deverá aliviar sua carga de trabalho, promovendo uma gestão mais organizada e eficiente, o que impactará diretamente na sua rotina e permitirá um foco maior nas atividades práticas e estratégicas da ONG.

Além dos atores internos, a ONG também possui *stakeholders* externos que influenciam e são impactados pelas operações da organização. Entre os *stakeholders* mais importantes estão os adotantes dos animais, que necessitam informações claras e atualizadas sobre os animais disponíveis para adoção. A comunicação com esse grupo é essencial para o sucesso das feiras de adoção, por exemplo, e a plataforma proposta ajudará a

centralizar e automatizar essas interações.

Outros *stakeholders* externos incluem possíveis patrocinadores e doadores, que demandam transparência nas operações financeiras da ONG. A plataforma também permitirá a prestação de contas mais eficiente, possibilitando que os recursos arrecadados sejam acompanhados de forma clara e objetiva.

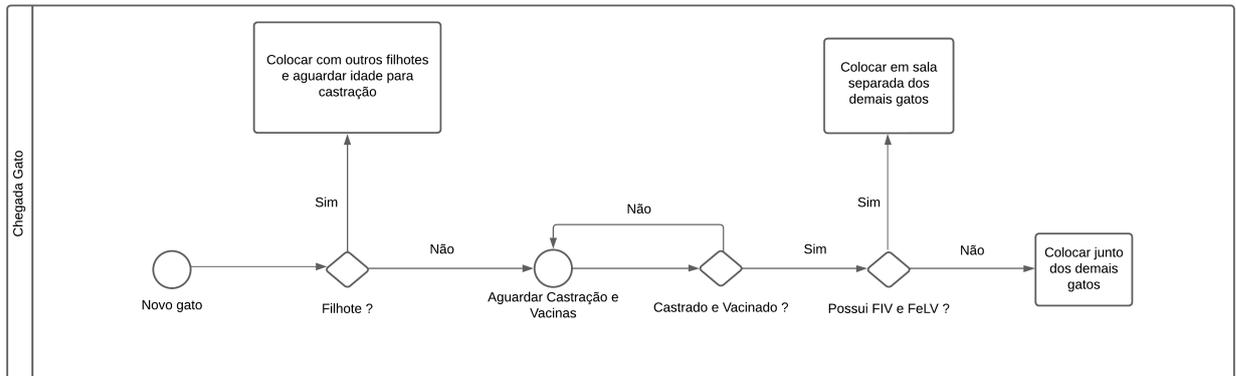
Por fim, há também a comunidade local e outros parceiros institucionais que podem se beneficiar diretamente da atuação da ONG. A colaboração entre ONGs em nível municipal, por exemplo, poderia ser facilitada por meio da integração da plataforma com sistemas municipais, permitindo um compartilhamento mais eficiente de informações e recursos.

O levantamento dos atores e stakeholders, juntamente com a análise do contexto da ONG Anjos na Terra, foi fundamental para o desenvolvimento de uma visão clara sobre os requisitos do sistema. A plataforma proposta visa atender às necessidades da ONG, proporcionando uma gestão mais eficiente e centralizada. Além disso, ela permitirá que a ONG se expanda e se organize melhor, resultando em um impacto positivo mais significativo para os animais resgatados e a comunidade envolvida.

### 3.2.2 Análise do Modelo de Negócio

A análise do modelo de negócio da ONG Parceira Anjos Na Terra foi feita com o auxílio da criação de diagramas em BPMN (Business Process Modeling Notation) um padrão de modelagem criado para ser uma forma de "rascunho universal" que pudesse ser compreendido por diversos analistas de negócio (WHITE, 2004) usando um conjunto de regras e conjunto de objetos definidos para a modelagem de fluxos de negócio. Com isso, é possível ter uma visão mais sistemática do funcionamento da organização, e em que pontos o sistema a ser desenvolvido pode atuar. Os fluxos a serem analisados serão: o processo de resgate do animal, o processo de adoção do animal, o processo de gestão de voluntários e captação de recursos, o processo de administração da ONG e o processo de criação de eventos.

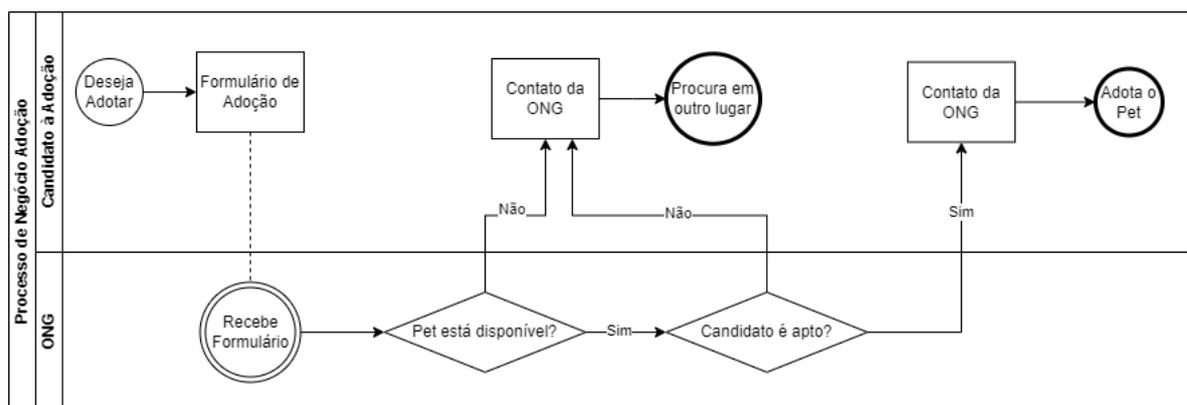
Figura 9 – Diagrama BPMN do Processo de Resgate do Animal



Fonte: BRITO, COHEN, PADULA 2024

A partir da análise do processo de resgate do animal, nota-se a necessidade de um sistema de cadastro das características do animal, como idade, vacinação e necessidades especiais. Além de um sistema para identificação visual e da localização do animal dentro da ONG. Uma solução interessante seria o uso de identificadores únicos para cada animal, que pode ser utilizado junto a um QR Code na coleira. Com isso, torna-se mais fácil a divulgação de quais animais estão disponíveis e prontos para a adoção e presença em evento, além de facilitar o trabalho dos voluntários na ONG para identificar cada pet.

Figura 10 – Diagrama BPMN do Processo de Adoção do Animal

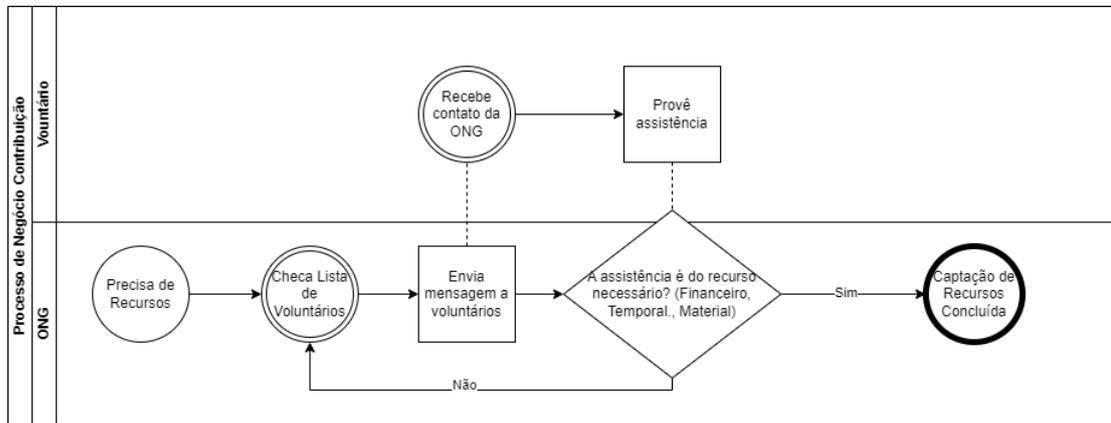


Fonte: BRITO, COHEN, PADULA 2024

A partir da compreensão do processo de adoção, percebe-se a relevância de um sistema de gestão de formulários padronizados, que coletam informações essenciais sobre a aptidão do candidato à adoção do animal como renda, profissão, número de moradores na

residência e afins. Além de um sistema de gerenciamento sobre os candidatos à adoção de cada pet, para fins de verificação da disponibilidade.

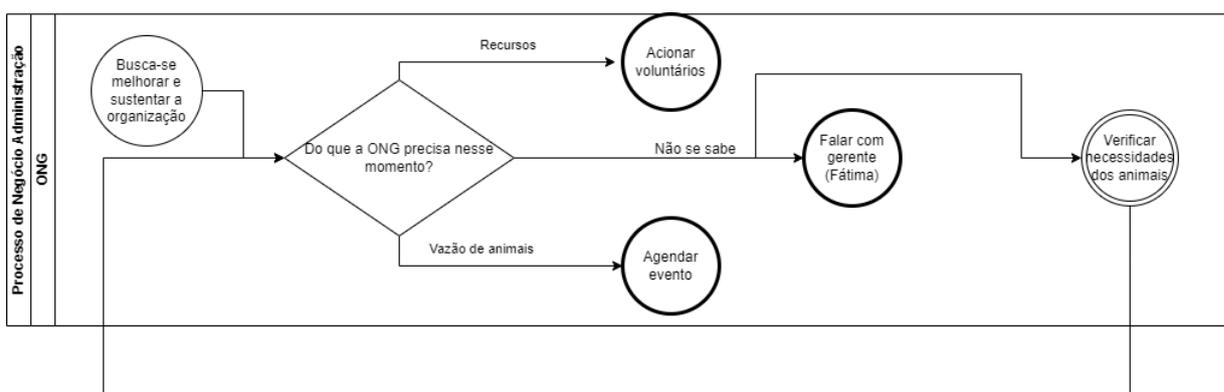
Figura 11 – Diagrama BPMN do Processo de Gestão de Voluntários e Captação de Recursos



Fonte: BRITO, COHEN, PADULA 2024

Analisando o processo de gestão de voluntários surge a necessidade de um cadastro e gerenciamento dos contatos dos voluntários, além do histórico de contribuição de cada um. Com isso, torna-se mais fácil a captação de recursos mais urgentes, sabendo-se qual tipo de recurso (material, financeiro ou de tempo) cada voluntário costuma contribuir.

Figura 12 – Diagrama BPMN do Processo de Administração da ONG

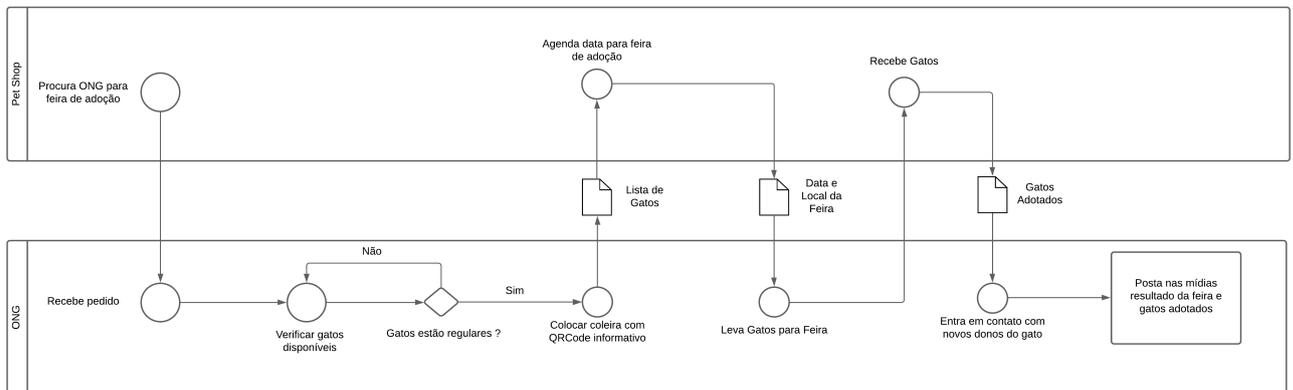


Fonte: BRITO, COHEN, PADULA 2024

No processo de administração da ONG nota-se um problema grave de distribuição das informações. Com isso, percebe-se a relevância de um sistema de fácil acesso para percepção do que é mais importante para a organização em dado momento, além de

gerenciar mais facilmente a necessidade dos animais. Um sistema de análise financeira e material poderia simplificar muito o processo.

Figura 13 – Diagrama BPMN do Processo de Criação de Eventos



Fonte: BRITO, COHEN, PADULA 2024

Por fim, na criação de eventos transparece a importância de um método de verificação da disponibilidade e aptidão dos animais da ONG à adoção, além de um sistema de identificação dos animais. A identificação torna-se solucionável a partir da utilização do QR Code único para cada animal, que pode ser escaneado pelos candidatos à adotante para ver mais informações sobre o pet e dar início ao processo de adoção e preenchimento dos formulários. Não obstante, um sistema de gerenciamento de eventos, similar a um calendário, para definição de quais pets irão para qual evento em qual data torna-se muito relevante.

### 3.2.3 Definição de Requisitos na instância de resgate e adoção de animais

Após conversas com os representantes da ONG e análise do seu funcionamento organizacional, foram levantados os principais requisitos funcionais e não funcionais que a plataforma deve atender para que seja possível não só organizar as atividades atuais da ONG, mas também permitir evoluções incrementais.

#### 3.2.3.1 Requisitos Funcionais

1. Identificação de usuários;
2. Diferentes níveis de permissionamento;
3. Cadastro de ONGs;

4. Cadastro de animais;
5. Geração de QR code por animal;
6. Edição de animais;
7. Alteração de status do animal (não adotado → adotado);
8. Listagem de animais (adotados e não adotados, por ONG);
9. Formulário de candidatos a adoção;
  - preenchimento por candidato;
  - visualização por responsáveis da ONG;
  - possibilidade de o candidato escolher um animal específico;
  - possibilidade de o candidato escolher características do animal que deseja (ex: se gato/cachorro, adulto/filhote, etc);
10. Listagem de todos os candidatos a adoção;
11. Listagem de candidatos a adoção por animal;
12. Alteração de status de candidatura de adoção (não adotado → adotado);
13. Cadastro de voluntários;
14. Listagem de voluntários;
  - todos;
  - por tipo de dedicação à ONG (tempo, dinheiro, materiais, lar temporário, etc);
15. Edição de voluntário;
16. Cadastro de registros financeiros;
17. Compilação de registros financeiros;
  - cálculo de receita;
  - cálculo de despesas;
  - cálculo de resultado;
  - gráfico de fluxo de caixa;
  - gráfico de distribuição de gastos;
  - gráfico de fontes de arrecadação;
  - gráfico de distribuição média;

18. Cadastro de eventos (feiras de adoção);
19. Registro de quais animais foram levados ao evento;
20. Registro de quais animais foram adotados no evento;
21. Registro de quais animais retornaram do evento.

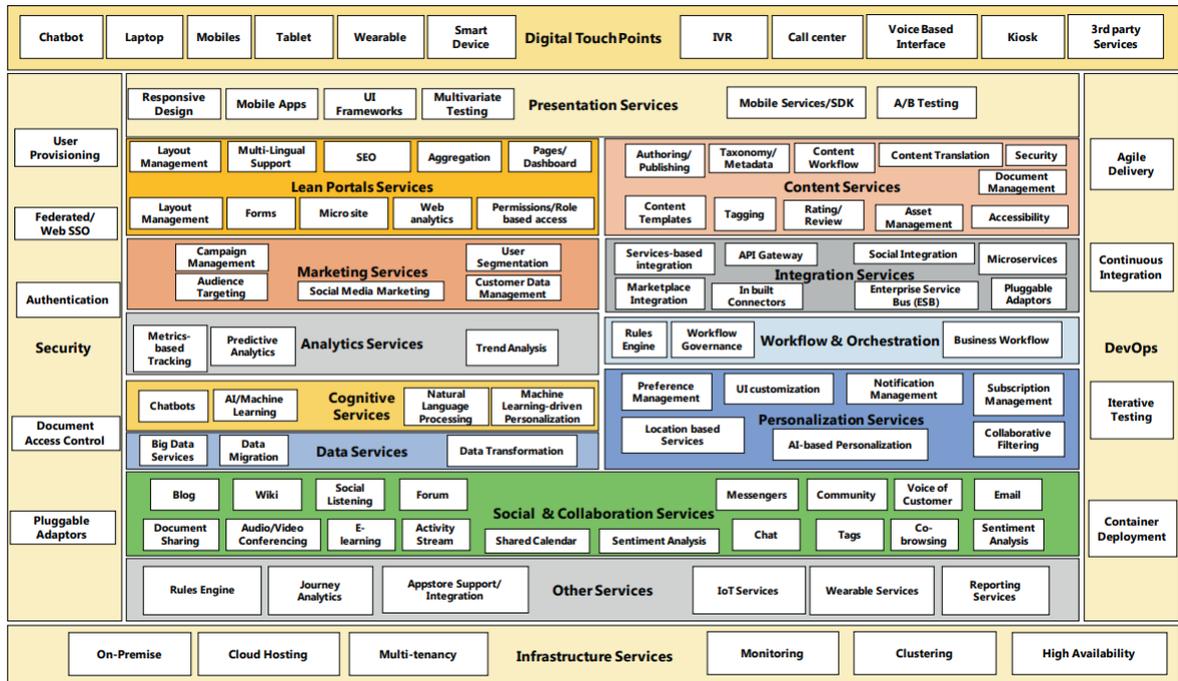
### 3.2.3.2 Requisitos Não Funcionais

1. *Design mobile-first*;
2. Arquitetura em microsserviços;
3. Banco de dados relacionais para entidades com dados bem estruturados (ex: pessoas);
4. Banco de dados não relacionais para entidades não estruturadas (ex: formulários de candidatos à adoção).

## 3.3 Arquitetura da Plataforma Digital de Sistemas de Administração para ONGs

Com base nas melhores práticas descritas na literatura sobre plataformas digitais robustas e escaláveis, a proposta para uma plataforma destinada à administração de ONGs deve incorporar um conjunto de requisitos organizados em camadas e funcionalidades específicas. Estes requisitos são fundamentais para garantir o alinhamento com os objetivos operacionais das ONGs, enquanto promovem eficiência, transparência e personalização.

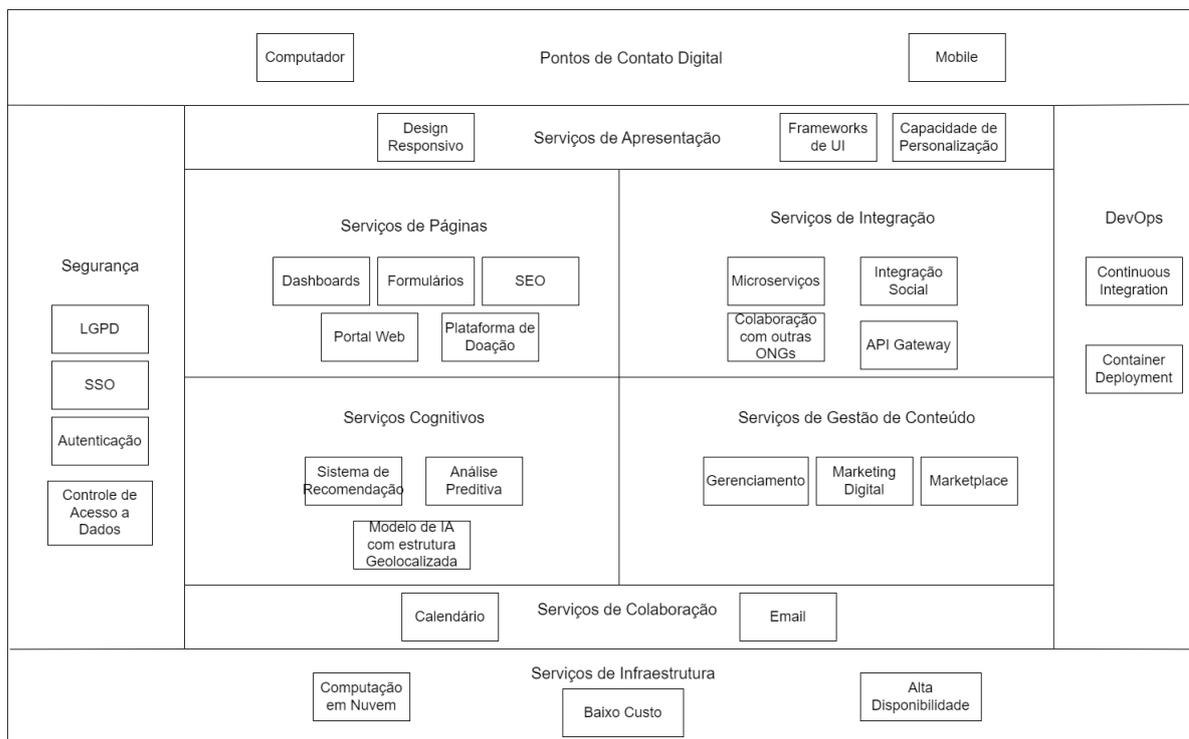
Figura 14 – Arquitetura de Plataforma Digital de Referência



Fonte: (SHIVAKUMAR; SETHI, 2019), Página 6

Com base na referência, foi possível projetar a seguinte arquitetura para a plataforma digital:

Figura 15 – Arquitetura de Plataforma Digital Projetada



Fonte: BRITO, COHEN, PADULA 2024

### 3.3.1 Plataforma Orientada a Módulos e Integração

- **Arquitetura Modular:**

- Construída sobre um modelo de microsserviços, permitindo a adição ou remoção de funcionalidades sem impacto no sistema.
- Suporte para extensibilidade futura, permitindo o crescimento e adaptação às mudanças no ambiente tecnológico.

- **Integração com Sistemas Externos:**

- APIs padronizadas para integração com plataformas municipais, redes sociais, sistemas de pagamento e dispositivos IoT.
- Adaptação a serviços baseados em REST e suporte para formatos como JSON e XML.

### 3.3.2 Funcionalidades de Apresentação e Experiência do Usuário

- **Interfaces Responsivas e Intuitivas:**

- Design responsivo para garantir acesso otimizado em desktops, tablets e smartphones.
- Personalização do layout para atender a diferentes públicos e idiomas.
- **Experiência Multicanal:**
  - Integração de todos os pontos de contato do usuário, incluindo portais web, aplicativos móveis e dispositivos IoT.
  - Acesso centralizado a informações relevantes e navegação uniforme.
- **Capacidades de Personalização:**
  - Personalização com base no contexto da ONG, como proteção animal, inclusão social e assistência a idosos.
  - Configuração individual para necessidades específicas, como visualizações de dados, relatórios ou notificações.

### 3.3.3 Gestão de Conteúdo e Campanhas

- **Gerenciamento de Conteúdo:**
  - Ferramentas para criação, publicação e tradução de conteúdo.
  - Fluxos de trabalho automatizados para revisão e aprovação.
- **Campanhas e Marketing Digital:**
  - Criação e gestão de campanhas de arrecadação e conscientização.
  - Integração com plataformas de redes sociais para aumentar o alcance.
  - Segmentação de público com base em dados demográficos e comportamentais.

### 3.3.4 Automação e Recomendações Baseadas em IA

- **Sistemas de Recomendação:**
  - Matchmaking entre pets e adotantes com base em compatibilidade.
  - Alocação automatizada de voluntários para atividades específicas, considerando habilidades e disponibilidade.
- **Análises Preditivas:**
  - Identificação de tendências para otimização de campanhas e alocação de recursos.
  - Sugestões de melhorias baseadas em padrões identificados nos dados históricos.

### 3.3.5 Segurança e Conformidade

- **Autenticação e Controle de Acesso:**
  - Suporte para SSO (Single Sign-On), LDAP e integrações com protocolos de segurança modernos, como OAuth e SAML.
  - Controle de acesso baseado em permissões para diferentes níveis de usuários.
- **Conformidade com Regulamentações:**
  - Estrutura preparada para atender às regulamentações locais e internacionais, como LGPD ou GDPR, garantindo a privacidade dos dados.

### 3.3.6 Suporte a Escalabilidade e Alta Disponibilidade

- **Infraestrutura Escalável:**
  - Suporte a implantações em nuvem, containers (Docker, Kubernetes) e ambientes híbridos.
  - Multitenancy para gerenciar múltiplas ONGs em uma única instância do sistema, garantindo isolamento de dados.
- **Alta Disponibilidade e Resiliência:**
  - Monitoramento em tempo real e estratégias de recuperação de desastres.
  - Redundância de sistemas para evitar falhas e perda de dados.

### 3.3.7 Análise e Relatórios

- **Painéis de Controle:**
  - Visualização centralizada de métricas de desempenho, impacto social e engajamento.
  - Relatórios configuráveis para diferentes públicos.
- **Indicadores de Impacto:**
  - Métricas específicas para cada ONG, como adoções concluídas, recursos arrecadados ou atendimentos realizados.
  - Relatórios preditivos para auxiliar na tomada de decisões estratégicas.

### 3.3.8 Sistemas de E-commerce e Marketplace

- **Marketplace Integrado:**

- Venda de produtos e serviços relacionados à ONG, como coleiras, artesanato ou serviços de consulta veterinária.
- Gerenciamento de estoque e integração com plataformas de e-commerce existentes.

- **Plataforma de Doação:**

- Portais personalizados para doações financeiras e em itens.
- Sistemas automatizados de geração de recibos e notificações.

### 3.3.9 Capacidades Colaborativas

- **Ferramentas de Colaboração:**

- Fóruns, chats, wikis e calendários para engajamento entre voluntários, beneficiários e administradores.
- Integração com sistemas de mensagens instantâneas e redes sociais.

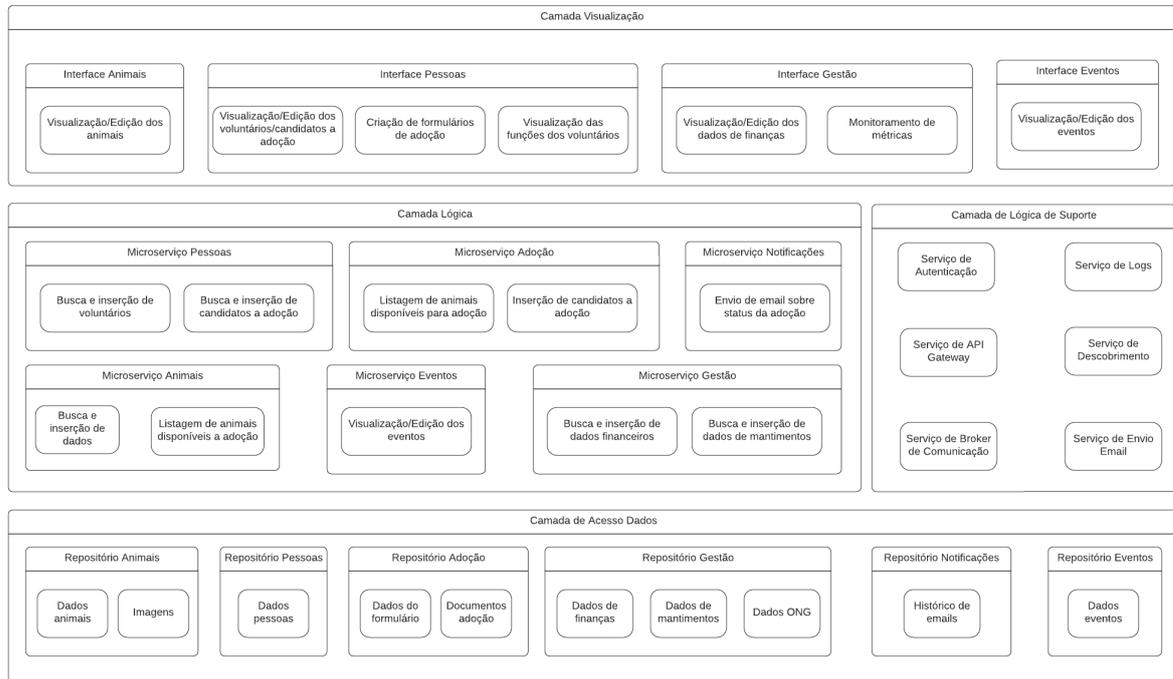
- **Parcerias Estratégicas:**

- Módulo para gerenciar parcerias com empresas, incluindo monitoramento de doações, voluntariado corporativo e campanhas conjuntas.
- Relatórios personalizados para empresas parceiras.
- Módulo orientado a comunicações entre ONGs para realocação de recursos

## 3.4 Arquitetura em Camadas da Instância de Resgate e Adoção de Pets

A Arquitetura adotada no projeto é o modelo de arquitetura em camadas, onde o projeto é dividido em 5 partes: Camada de Negócios, Engenharia, Informação, Computação e Interface. (SHIVAKUMAR; SETHI, 2019; RICHARDS, 2015)

Figura 16 – Camadas de Arquitetura do Sistema

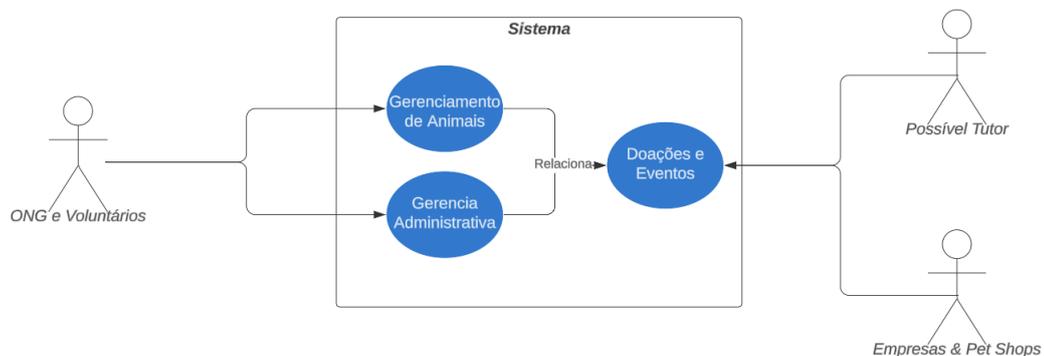


Fonte: BRITO, COHEN, PADULA 2024

### 3.4.1 Camada de Negócios

A camada de negócios descreve como os usuários interagem com o sistema. Neste caso, trata-se da análise dos processos de negócio feita nas seções anteriores, de maneira que o sistema atue para automatizar e regularizar esses processos dentro de um modelo, tal que os processos de negócio da organização tornem-se mais estruturados e similares a maneira como foram descritos nos diagramas BPMN.

Figura 17 – Diagrama de funcionalidades do sistema e usuários



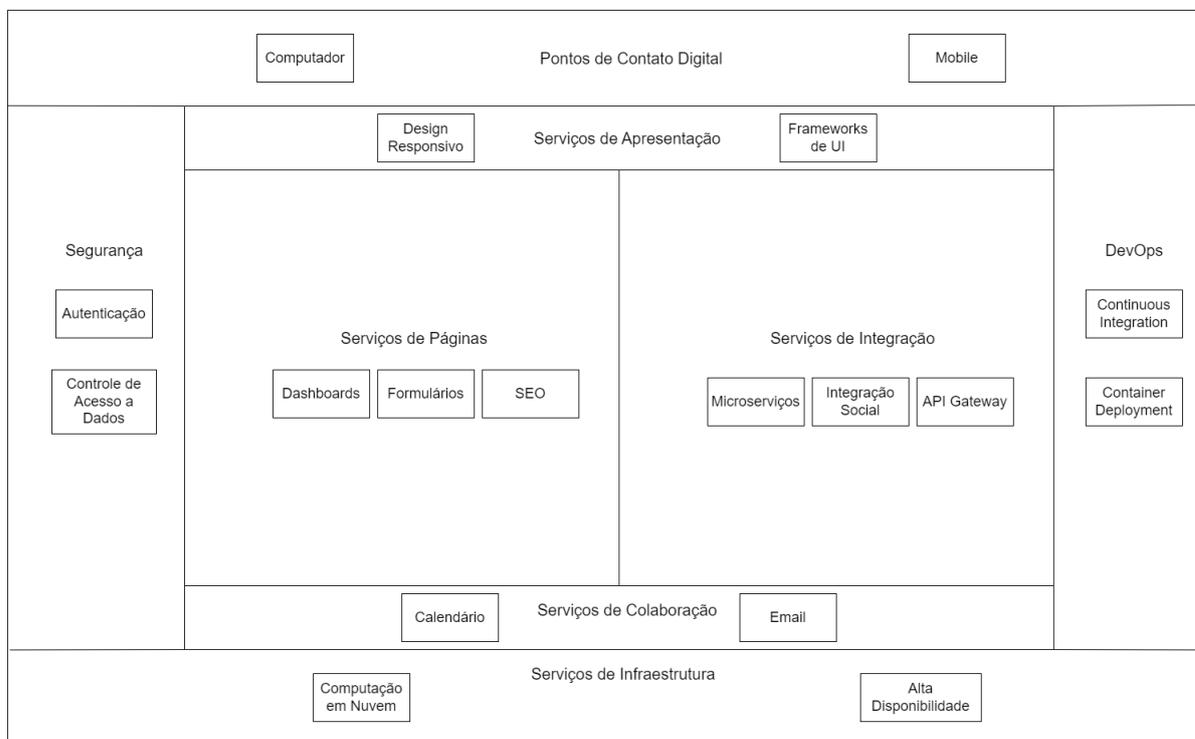
Fonte: BRITO, COHEN, PADULA 2024

Com o levantamento de requisitos, estudo dos atores no sistema e do modelo de negócio é possível definir uma interação entre empresas, Ong e voluntários e possíveis tutores para os animais na instância proposta.

### 3.4.2 Camada de Engenharia

O diagrama proposto para a arquitetura da plataforma digital apresenta uma série de serviços essenciais que sustentam a experiência do usuário e a funcionalidade do sistema. A camada de engenharia do sistema toma como referência a arquitetura ideal proposta em (SHIVAKUMAR; SETHI, 2019). Nessa seção iremos analisar a camada de engenharia da plataforma proposta, passando por cada um dos serviços propostos.

Figura 18 – Camada de Engenharia da Instância em Resgate e Adoção de Pets



Fonte: BRITO, COHEN, PADULA 2024

- Pontos de Contato do Usuário:** A camada de pontos de contato é composta por dispositivos como "Computador" e "Mobile", que representam os canais de acesso dos usuários. Esses dispositivos permitem uma experiência integrada, onde o usuário pode acessar a plataforma de qualquer lugar e em qualquer dispositivo. Esse nível de acessibilidade é crucial para a experiência dos usuários para que possam acessar a plataforma da maneira que for possível.

- **Serviços de Apresentação:** são os componentes que garantem que a plataforma seja visualmente consistente e intuitiva em diferentes dispositivos. Com o design responsivo, o layout e o conteúdo da plataforma ajustam-se para proporcionar uma experiência uniforme e amigável tanto em dispositivos móveis quanto em desktops. Enquanto utilizar frameworks de UI permite uma implementação mais ágil e estruturada da interface. Frameworks de desenvolvimento web oferecem componentes reutilizáveis e funcionais que ajudam a construir interfaces complexas de forma modular e eficiente, promovendo uma experiência de uso consistente e de alta qualidade.
- **Serviços de Portal:** São os componentes que enriquecem a experiência do usuário e facilitam a navegação e a visibilidade da plataforma. Os formulários são usados para coletar informações dos usuários e animais, como dados de registro e adoção. Eles são componentes essenciais para plataformas que exigem input de dados dos usuários, e precisam ser seguros e de fácil preenchimento para promover uma boa experiência, enquanto o "SEO"(Otimização para Mecanismos de Busca) ajuda a melhorar a visibilidade da plataforma nos motores de busca, como Google e Bing. Técnicas de SEO incluem otimização de palavras-chave, metadados, e práticas que tornam o site mais amigável para buscas. Um bom SEO atrai mais usuários para a plataforma, aumentando o alcance e relevância. Já os dashboards são painéis informativos que centralizam dados importantes para o usuário, permitindo uma visão panorâmica de informações e métricas relevantes. Esses painéis são muito úteis para gerenciar e monitorar dados em tempo real, auxiliando na tomada de decisões rápidas e embasadas.
- **Serviços de Integração:** são os componentes fundamentais para a integração da plataforma com outros serviços e sistemas. A arquitetura de microserviços divide a aplicação em serviços menores e independentes que se comunicam entre si. Isso facilita a escalabilidade e a manutenção, pois cada serviço pode ser atualizado, substituído ou escalado individualmente sem impactar o sistema inteiro. Além disso, permite uma divisão clara de responsabilidades, melhorando a organização e a agilidade do desenvolvimento. A integração social permite que a plataforma se conecte com redes sociais (como Facebook, Instagram, Whatsapp), proporcionando funcionalidades de compartilhamento, e interação com redes externas. Isso facilita o engajamento dos usuários e amplia o alcance da plataforma, promovendo uma experiência mais conectada. O API Gateway funciona como um ponto central de entrada para todas as solicitações externas que chegam à plataforma. Ele facilita a integração com serviços externos, oferece roteamento e segurança, e otimiza o tráfego de dados. Com um gateway bem configurado, a plataforma pode expor APIs de forma segura e escalável, além de melhorar a performance ao gerenciar o fluxo de solicitações.

- **Serviços de Infraestrutura:** são os serviços que sustentam a infraestrutura da plataforma. A alta disponibilidade é essencial para garantir que a plataforma continue operando mesmo em caso de falhas no sistema. Esse serviço inclui práticas como replicação de dados, balanceamento de carga e failover, que aumentam a resiliência da plataforma. A alta disponibilidade é crucial para evitar downtime e garantir uma experiência de uso contínua. Isso pode ser alcançado usando serviços de computação em nuvem, que permitem a alocação de servidores de maneira dinâmica em plataformas como a AWS. A computação em nuvem permite que a plataforma utilize recursos escaláveis e flexíveis, além de facilitar o armazenamento e processamento de dados. Com serviços de nuvem, a plataforma pode escalar conforme a demanda, sem necessidade de infraestrutura física.
- **Serviços de Segurança:** são os componentes que asseguram a proteção dos dados dos usuários e a segurança das transações realizadas na plataforma. A autenticação impede acessos não autorizados, e o controle de acesso permite a gestão de permissões de acesso a diferentes áreas da plataforma. Com um bom controle de acesso, a plataforma pode definir quais usuários têm permissão para visualizar, modificar ou excluir dados específicos, garantindo a segurança e a privacidade das informações.
- **Serviços de Colaboração e Comunicação:** são os componentes externos que quando integrados à plataforma automatizam processos de organização e mensagens. O calendário é um componente que facilita a organização e o agendamento de eventos. Já a automatização de emails e notificações permitem uma comunicação mais eficiente.
- **Serviços de DevOps:** São serviços que suportam a metodologia ágil e promovem melhorias contínuas, garantindo que a plataforma esteja sempre atualizada e alinhada com as necessidades do negócio. O uso de contêineres, como Docker e Kubernetes, permite que a plataforma seja implantada em ambientes isolados e consistentes, facilitando o gerenciamento e a escalabilidade dos serviços. Contêineres tornam o processo de deployment mais flexível e robusto, permitindo que a aplicação seja executada em qualquer ambiente sem problemas de compatibilidade. Já a integração contínua é uma prática que permite que desenvolvedores integrem código novo com frequência, verificando a compatibilidade e a estabilidade do sistema a cada alteração.

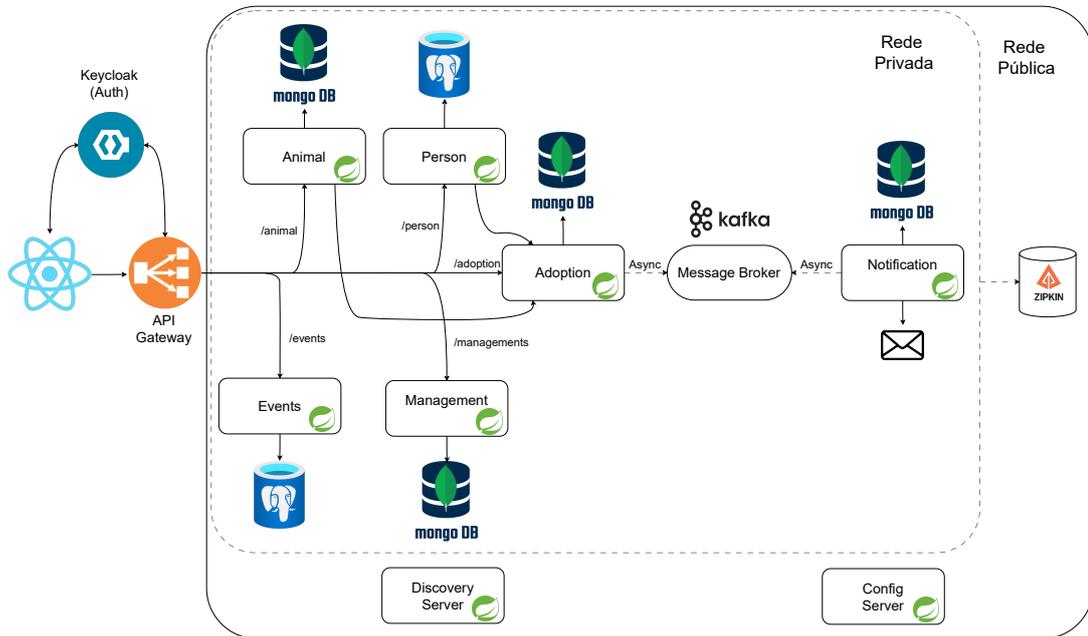
### 3.4.3 Camada de Computação

Na camada de Computação, é definida uma arquitetura em microserviços, integrando várias funcionalidades e serviços distribuídos. Abaixo está uma descrição dos principais componentes e da interação entre eles:

- **Front-end e API Gateway:** A aplicação front-end (representada pelo ícone do React) interage com o backend através de um API Gateway. O API Gateway age como uma entrada unificada, encaminhando as requisições para os microserviços apropriados. Ele está localizado na "Rede Pública", o que permite o acesso dos usuários externos ao sistema.
- **Microserviços Principais:** Animal, Person, Events, Finance, Adoption e Notification: São microserviços individuais que representam funcionalidades específicas do sistema. Cada um deles possui sua própria base de dados (MongoDB, PostgreSQL, etc.), garantindo a independência e a coesão dos serviços.
  - Animal e Person: Esses endpoints permitem a manipulação de dados relacionados a animais e pessoas.
  - Events: Responsável pelo gerenciamento de eventos.
  - Finance: Lida com informações financeiras, deverá registrar e monitorar transações relacionadas.
  - Adoption: Cuida do processo de adoção, vinculando dados de pessoas e animais.
  - Notification: Serviço de notificação que enviará comunicações via e-mail ou outros meios quando acionado.
- **Broker de Mensagens (Kafka):** Um Message Broker (Kafka) é utilizado para comunicações assíncronas entre os microserviços, especialmente entre o serviço de adoção e o serviço de notificação. Isso permite o processamento de eventos de forma desacoplada, onde, por exemplo, ao registrar uma adoção, uma mensagem é publicada e consumida pelo serviço de notificação para enviar atualizações aos usuários.
- **Serviços de Configuração e Descoberta:** Discovery Server e Config Server (ambos usando Spring): Esses servidores fornecem serviços de descoberta (Eureka) e configuração centralizada (Config Server), essenciais para uma arquitetura de microserviços escalável e facilmente configurável.
- **Ferramenta de Monitoramento e Log (Zipkin):** Utilizado para rastreamento distribuído. Ele ajuda a monitorar e rastrear requisições que percorrem vários microserviços, facilitando a identificação de gargalos ou problemas no fluxo de dados.

Esse sistema é altamente modular e escalável, onde cada microserviço é autônomo, mas comunica-se com outros quando necessário, tanto por meio de requisições diretas via API quanto por comunicação assíncrona via Kafka. A presença de Discovery e Config Server indica que o sistema foi projetado para ser configurável e facilmente adaptável, enquanto o Zipkin facilita o monitoramento e manutenção do sistema em produção.

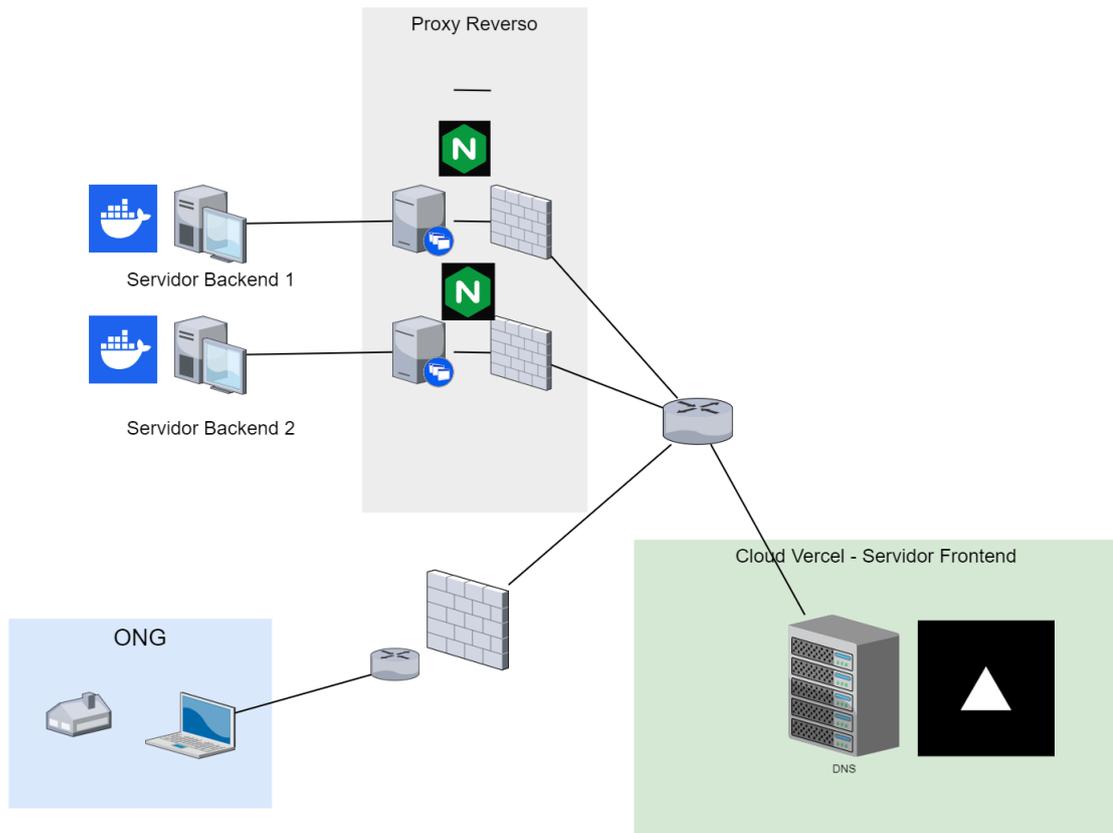
Figura 19 – Camada de Computação em Microserviços



Fonte: BRITO, COHEN, PADULA 2024

Cada domínio, na arquitetura em microserviços pode ser operado por máquinas diferentes, para tanto, foi feita a conteneurização de cada módulo, e a distribuição dos mesmos em um sistema descentralizado:

Figura 20 – Infraestrutura da Implementação



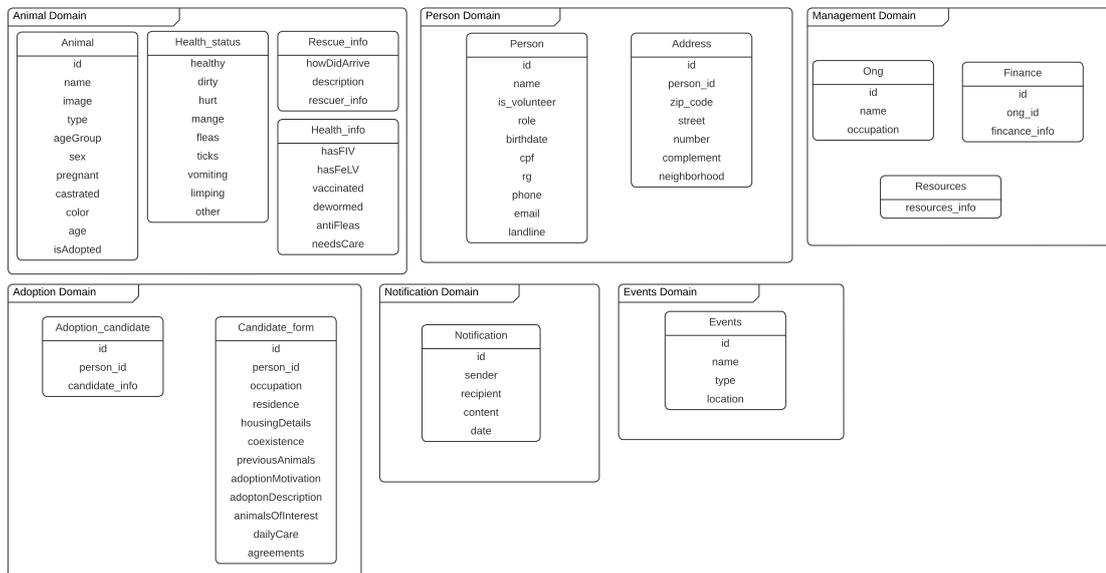
Fonte: BRITO, COHEN, PADULA 2024

A infraestrutura compõe 2 servidores backend com os módulos em docker distribuídos entre si, assegurados por um proxy-reverso; junto com um servidor frontend carregado na plataforma Vercel de distribuição em nuvem.

#### 3.4.4 Camada de Informação

Na camada de informação são definidas as estruturas de dados presentes para armazenamento de informações no sistema. Para cada um dos serviços da camada de computação foi definido um domínio, no qual as entidades dos banco de dados são arquitetadas, independentemente se o banco é estruturado (SQL) ou não (NoSQL).

Figura 21 – Entidades dos Bancos de dados na Camada de Informação



Fonte: BRITO, COHEN, PADULA 2024

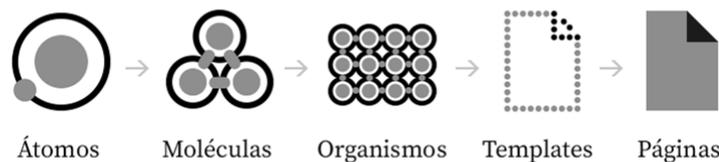
- **Domínio Animal:** Contém informações relacionadas aos animais.
  - **Animal:** Inclui campos como *id*, *nome*, *sexo*, *idade*, *data de criação* e outros atributos.
  - **Health\_status:** Representa o estado de saúde do animal, com campos como *id*, *animal\_id* e um campo genérico para informações adicionais.
- **Domínio Pessoa:** Armazena dados das pessoas envolvidas.
  - **Person:** Armazena informações pessoais, como *id*, *nome*, *é voluntário*, *papel*, *data de nascimento*, *tipo de documento*, *número do documento*, *telefone*, *email* e *data de criação*.
  - **Address:** Contém informações de endereço, incluindo *id*, *person\_id*, *código postal*, *rua*, *número*, *complemento* e *bairro*.
- **Domínio Eventos:** Define eventos associados a pessoas ou animais.
  - **Events:** Inclui campos como *id*, *nome*, *tipo* e *localização* para descrever diferentes eventos.
- **Domínio Adoção:** Contém informações sobre candidatos à adoção.
  - **Adoption\_candidate:** Define candidatos à adoção com *id*, *person\_id* e um campo genérico para outras informações.

- **candidate\_form:** Representa formulários de candidatos com *id*, *person\_id* e um campo genérico para dados adicionais.
- **Domínio de Gestão:** Gerencia informações financeiras, de recursos e de organização.
  - **Ong:** Representa uma organização com *id*, *nome* e *ocupação*.
  - **Finance:** Contém dados financeiros, incluindo *id*, *ong\_id* e um campo específico.
  - **Resources:** Um campo genérico para gerenciar recursos financeiros ou materiais.
- **Domínio de Notificação:** Gerencia notificações entre entidades.
  - **Notification:** Inclui campos como *id*, *remetente*, *destinatário*, *conteúdo* e *data* para comunicação entre entidades.

### 3.4.5 Camada de Interface

A arquitetura do Frontend segue as heurísticas de UX Design (NIELSEN, 1994), utilizando-se do conceito de Design Atômico (QUEVEDO, 2023) para garantia de modularidade. O design atômico permite que componentes da interface sejam desenvolvidos separadamente, começando pelos átomos (botões, campos de texto, etc), seguindo para as moléculas (como formulários e grupos de texto), organismos (cartões, seções de login, etc), templates (páginas básicas sem aplicação de contexto), até a implementação final das páginas instanciando os componentes descritos.

Figura 22 – Camada de Interface em Design Atômico



Fonte: BRITO, COHEN, PADULA 2024

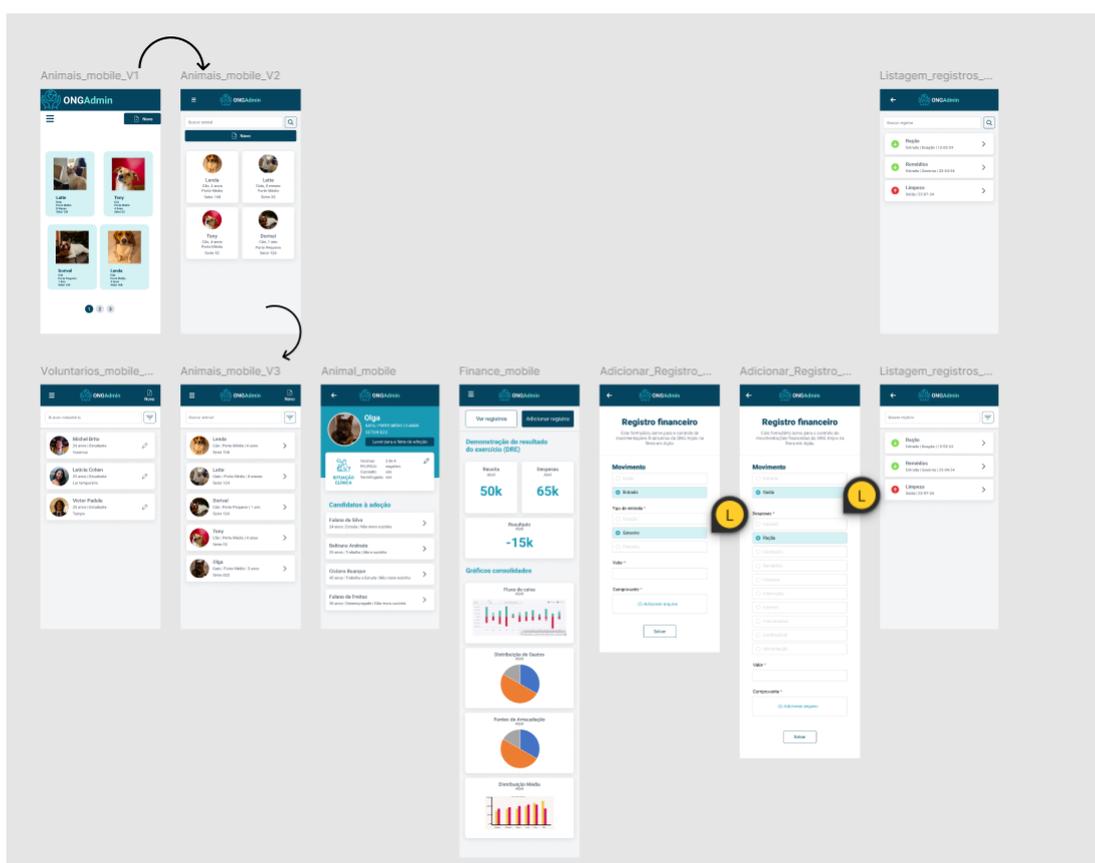
Dentre as heurísticas de UX Design, vale citar o foco nos seguintes tópicos:

- 1. Visibilidade do status do sistema
- 2. Compatibilidade do sistema com o mundo real
- 3. Controle e liberdade do usuário

- 4. Consistência e padrões
- 5. Prevenção de Erros
- 6. Reconhecimento ao invés de Lembrança
- 7. Flexibilidade e Eficiência de Uso
- 8. Design Estético e Minimalista
- 9. Facilidade para os usuários recuperarem de erros
- 10. Ajuda e Documentação

Com isso foi possível definir uma interface para a instância a ser implementada conforme o diagrama:

Figura 23 – Prototipação da Camada de Interface desenvolvido no Figma



Fonte: BRITO, COHEN, PADULA 2024

## 4 Implementação

Neste capítulo são exploradas as tecnologias utilizadas na implementação, seguido do desenvolvimento e operação e por fim, os testes de integração do projeto.

### 4.1 Tecnologias Utilizadas

#### 4.1.1 Microserviços

São uma abordagem arquitetônica para desenvolver um sistema de software como um conjunto de serviços pequenos, independentes e interconectados. Em contraste com a abordagem monolítica, onde todo o aplicativo é desenvolvido como uma única unidade, os microserviços dividem a aplicação em componentes menores, cada um executando uma função específica.

Cada módulo é desenvolvido, implantado e gerenciado de forma independente. Assim permitindo maior flexibilidade e escalabilidade, entretanto, tal abordagem gera uma complexidade grande em relação à abordagem monolítica. Os diferentes microserviços devem ser capazes de se comunicar entre si por meio de APIs (Application Program Interface) que podem solicitar dados ou serviços de outros componentes.

#### 4.1.2 Design Patterns

Design Patterns são soluções reutilizáveis para problemas comuns encontrados no desenvolvimento de software. Eles não são soluções específicas, mas sim diretrizes gerais que podem ser aplicadas e adaptadas de acordo com o contexto e os requisitos do projeto. Neste caso iremos utilizar os padrões de Injeção de Dependências (DI - Dependency Injection) e Inversão de Controle (IoC - Inversion of Control).

- **Injeção de Dependência:** É um padrão de design utilizado principalmente em programação orientada a objetos para gerenciar as dependências entre componentes do software. Normalmente, um objeto precisa de outros para realizar suas operações. Em vez de criar essas dependências dentro dele mesmo, na DI, as dependências são fornecidas externamente. Isso geralmente é feito por meio de construtores, métodos ou propriedades.

Assim o código se torna mais modular e flexível para alterações futuras, já que os objetos não estão diretamente acoplados às suas dependências.

- **Inversão de Controle:** É um princípio de design de software que está relacionado à maneira como os componentes de um sistema interagem e dependem uns dos outros. Ao contrário do fluxo de controle tradicional, onde um componente controla a execução e a lógica de outros componentes, na inversão de controle, o comando da execução é transferido para um mecanismo externo. Em outras palavras, em vez de um componente chamar diretamente outros para realizar determinadas tarefas, ele delega o controle e a execução dessas tarefas para um framework. Ele, por sua vez, é responsável por gerenciar a criação e a interconexão dos componentes, assim como suas dependências.

A IoC é frequentemente implementada em conjunto com padrões como Injeção de Dependências e o uso de frameworks específicos de IoC, como Spring, para Java. Assim, permitindo uma arquitetura mais flexível e desacoplada, promovendo modularidade ao sistema.

### 4.1.3 Arquitetura MVC

A arquitetura MVC (Model-View-Controller) é um padrão de design que organiza aplicações em três camadas distintas. O Model representa a lógica de negócios e a estrutura de dados da aplicação, gerenciando as interações com o banco de dados e a manipulação de informações. Ele é responsável por garantir que as regras do negócio sejam aplicadas corretamente. A View é a camada que apresenta os dados ao usuário, exibindo a interface gráfica ou saída visual. Ela é responsável por renderizar as informações provenientes do Model e por capturar as interações do usuário, como cliques e inserção de dados.

O Controller atua como intermediário entre o Model e a View, recebendo as entradas da interface (View), processando-as, e, em seguida, chamando o Model para realizar as operações necessárias. Após a interação com o Model, o Controller atualiza a View para refletir as mudanças. Essa divisão facilita a manutenção e permite a escalabilidade, além de separar responsabilidades. Em uma arquitetura MVC é comum adicionar serviços para organizar a lógica de negócios que não se encaixam diretamente no Model. Os serviços servem como uma camada adicional que encapsula operações específicas e complexas, como integrações com APIs externas, processamento de dados ou outras tarefas que não devem ficar no Controller nem no Model.

### 4.1.4 Ecossistema Spring

O ecossistema Spring é um conjunto de projetos e ferramentas que facilitam o desenvolvimento de aplicações Java, fornecendo uma estrutura flexível e robusta para lidar com diversas necessidades de uma aplicação, como injeção de dependências, gerenciamento de transações e segurança. O Spring Framework, o núcleo desse ecossistema, utiliza o

conceito de inversão de controle (IoC) para gerenciar automaticamente as dependências entre os componentes, promovendo uma arquitetura modular e desacoplada.

Além do framework principal, o ecossistema inclui projetos especializados que atendem a necessidades específicas. O Spring Boot, por exemplo, simplifica a criação de aplicativos Spring, fornecendo uma configuração mínima e pronta para uso, com servidores embutidos e integração facilitada com bancos de dados e bibliotecas de terceiros. Outro componente popular é o Spring Data, que oferece abstrações para trabalhar com diferentes tecnologias de persistência, como bancos de dados relacionais. Além disso, o Spring Security e o Spring Cloud são exemplos de como o ecossistema também atende a requisitos avançados, como segurança de aplicativo e desenvolvimento de microsserviços. Enquanto o Spring Security oferece ferramentas para a autenticação e controle de acesso dos usuários, o Spring Cloud ajuda a desenvolver sistemas distribuídos e resilientes. Resultando, todos os componentes tornam o ecossistema do Spring uma excelente escolha para criar aplicativos corporativos que são escaláveis e seguros.

#### 4.1.5 JWT Token

JSON Web Token, em português Token de Json Web, é um padrão aberto que transmite informações estabelecidas entre duas partes no formato de um JSON objeto. JWT encontra ampla utilização na etapa de autenticação e autorização. Durante essa fase, o servidor emite um token de assinatura e o envia para o cliente, onde é armazenado em “cookies” ou localStorage. O token também contém um payload, que consiste em dados relacionados, como um identificador de usuário. Além disso, JWT pode ser verificado independentemente em relação ao servidor, o que o torna particularmente efetivo em sistemas distribuídos.

#### 4.1.6 Ecossistema React

React é uma biblioteca JavaScript focada em construir componentes reutilizáveis e reativos. o React permite que desenvolvedores de frontend criem interfaces dinâmicas e de alta performance de forma eficiente. O ponto central do React é o conceito de componentes, pequenas partes reutilizáveis da interface que encapsulam o comportamento e a aparência, promovendo modularidade e reusabilidade de código.

Um dos principais diferenciais do React é o uso do Virtual DOM, uma representação leve do DOM real, que possibilita atualizações de interface mais rápidas e eficientes. Ao invés de manipular diretamente o DOM, o React faz uma comparação entre o Virtual DOM e o DOM real, aplicando apenas as mudanças necessárias de forma otimizada. Isso resulta em uma melhora significativa na performance de aplicações, especialmente aquelas com interfaces complexas e grande volume de interações. Outra característica que faz o

React se destacar é a unidirecionalidade dos dados. Isso significa que o fluxo de dados é sempre de cima para baixo, ou seja, os dados fluem dos componentes pais para os componentes filhos. Além disso, os desenvolvedores tem controle do fluxo de estados e dados da página diretamente de componentes funcionais. Essa abordagem torna o estado e o comportamento da aplicação mais previsíveis, facilitando o rastreamento de bugs e a depuração do código.

#### 4.1.7 Framework NextJS

O framework NextJS é um conjunto de ferramentas construído para o ecossistema React, com o intuito de fornecer mais controle aos desenvolvedores de frontend sobre os servidores que fornecem o código e conteúdo de plataformas web aos clientes. Com isso, é possível a criação de interfaces com APIs externas de forma mais simplificada, além de possibilitar que parte do conteúdo seja renderizado ainda no servidor, o que possibilita carregamentos mais rápidos. Dessa maneira, não só é possível aliviar a carga de processamento das máquinas clientes como também melhorar a experiência do usuário. Além disso, com parte do conteúdo partindo diretamente do servidor torna-se mais viável o acesso do conteúdo por ferramentas e índices de busca, o que também traz mais visibilidade para a plataforma. Por fim, o Framework traz suporte a TypeScript, facilitando o desenvolvimento com tipagem estática e tornando o processo de construção de aplicações mais robusto e menos propenso a erros.

#### 4.1.8 Tailwind CSS

O Tailwind CSS é um framework de utilitários para CSS que permite aos desenvolvedores criar interfaces de usuário de forma rápida e eficiente. Diferentemente de frameworks tradicionais como o Bootstrap, que fornecem componentes pré-estilizados, o Tailwind oferece classes utilitárias de propósito único que podem ser combinadas para construir designs personalizados diretamente no markup HTML ou Componentes React/TSX. Ao adotar uma abordagem "utility-first", o Tailwind promove a consistência e a reutilização de estilos, reduzindo a necessidade de escrever CSS personalizado e minimizando problemas de escopo e especificidade. Além disso, o Tailwind facilita a criação de designs responsivos sem a necessidade de escrever media queries manualmente, pois suas classes podem ser aplicadas condicionalmente em diferentes pontos de interrupção. O Tailwind também oferece um sistema que remove classes não utilizadas no ambiente de produção, resultando em arquivos CSS menores e melhor desempenho do site. Com isso, o Tailwind se torna uma ferramenta muito útil na arquitetura proposta, uma vez que consegue garantir um design responsivo, e ao mesmo tempo consistente e flexível entre os módulos da plataforma.

### 4.1.9 Conteirização: Docker

A conteirização representa uma abordagem para a virtualização de aplicações e suas respectivas dependências, oferecendo isolamento e portabilidade em ambientes de execução. Ao contrário da virtualização tradicional, que utiliza máquinas virtuais (VMs) completas, cada uma contendo seu próprio sistema operacional, os contêineres compartilham o mesmo núcleo (kernel) do sistema operacional subjacente, otimizando o uso de recursos.

Ferramentas como o Docker desempenham um papel central nesse processo, permitindo a criação de contêineres a partir de imagens Docker, que encapsulam todo o ambiente necessário para a execução de uma aplicação, incluindo bibliotecas, dependências e variáveis de ambiente. O uso de contêineres garante a consistência entre diferentes ambientes de execução, resolvendo problemas de compatibilidade, uma vez que as imagens são imutáveis e podem ser executadas de forma idêntica em qualquer sistema compatível. O processo de construção de contêineres é tipicamente gerido por meio de arquivos Dockerfile, os quais definem, de maneira programática, os passos necessários para a criação da imagem Docker.

### 4.1.10 Integração e Desenvolvimento Contínuos (CI/CD)

A Integração e o Desenvolvimento Contínuos (CI/CD) são práticas fundamentais para garantir um fluxo de desenvolvimento ágil e automatizado. Através dessas práticas, o código é constantemente integrado e validado com a execução automática de testes (CI), seguido pela entrega ou implantação automatizada em ambientes de produção (CD). O objetivo é acelerar o ciclo de desenvolvimento, melhorar a qualidade do software e minimizar erros humanos. Essa automação torna-se ainda mais eficiente com o uso de contêineres, que encapsulam o código e suas dependências em um ambiente isolado e portátil, garantindo consistência desde o desenvolvimento até a produção.

Nos sistemas de frontend, a Vercel utiliza CI/CD de forma otimizada, integrando automaticamente mudanças feitas no código diretamente com o repositório Git. Além disso, com o uso de contêineres Docker, é possível garantir que as dependências e a configuração do ambiente sejam consistentes em todas as fases do ciclo de desenvolvimento. Cada vez que uma atualização no código é feita, o Docker ajuda a garantir que o ambiente de execução da aplicação permaneça idêntico ao ambiente de desenvolvimento, evitando problemas de compatibilidade. Isso torna o processo de deploy extremamente ágil, permitindo que o frontend seja atualizado em tempo real, de forma confiável.

No backend, a Google Cloud Platform (GCP) mostra-se como uma possível ferramenta a ser utilizada ao oferecer um conjunto de ferramentas que suporta o uso de contêineres em pipelines CI/CD, como o Cloud Build e Cloud Run. Com o Docker, é possível criar imagens de contêineres que encapsulam a aplicação backend, que então

podem ser implantadas automaticamente nos ambientes de produção. O uso de contêineres proporciona escalabilidade e facilita a gestão dos serviços, além de garantir que o backend esteja sempre sincronizado com as atualizações de frontend feitas na Vercel. Isso resulta em uma arquitetura robusta, onde o CI/CD orquestra todas as fases do desenvolvimento e implantação de maneira contínua e eficiente, simulando e preparando o sistema para o uso real.

## 4.2 Desenvolvimento e Operação

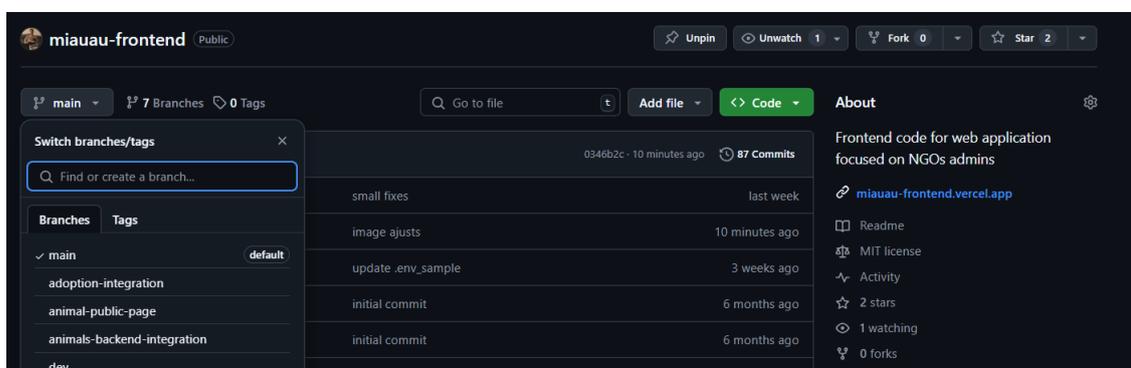
Nessa seção será explorada a implementação dos ambientes de execução do sistema e a operação do mesmo.

### 4.2.1 Ambientes de Desenvolvimento, Testes e Produção

Com a arquitetura e as ferramentas a serem utilizadas definidas, o desenvolvimento da implementação parte da criação de repositórios de código fonte com rastreamento de alterações usando o sistema Git. Um para o desenvolvimento de frontend, e outro para backend. Em seguida, para a colaboração da equipe de desenvolvimento, instanciação dos sistemas em ambiente de produção e publicação do código desenvolvido em licenças de software livre como MIT, o código foi carregado na plataforma Github.

Com isso, cada desenvolvedor pode fazer alterações em uma estrutura hierárquica, separando os ambientes de desenvolvimento, testes e produção pelas branches de cada feature, seguindo para a dev e enfim na branch principal. Cada avanço nessa hierarquia passa por testes manuais e automatizados como controle de qualidade.

Figura 24 – Repositório de Código Fonte para Frontend



Fonte: BRITO, COHEN, PADULA 2024

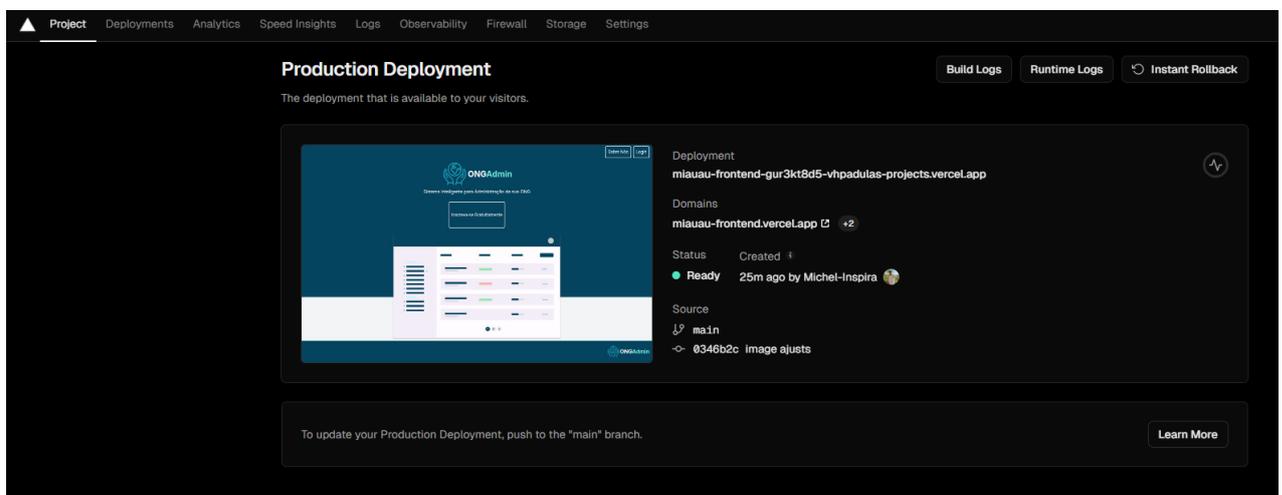
Em ambiente de produção, o software pode ser containerizado para instanciamento em ambiente de produção. As instâncias containerizadas podem então ser carregadas em

servidores locais, ou usando ferramentas de nuvem como GCP e AWS.

No caso dessa primeira implementação, os serviços foram instanciados seguindo essa infraestrutura apresentada na Figura 20.

Há uma distribuição dos módulos dos microserviços em diferentes máquinas utilizando uma solução de baixo custo energético e financeiro. Os ambientes de produção contam com integração contínua das mudanças do código fonte, permitindo um processo de desenvolvimento iterativo conforme o feedback da ONG.

Figura 25 – Monitoramento do Ambiente de Produção em Nuvem

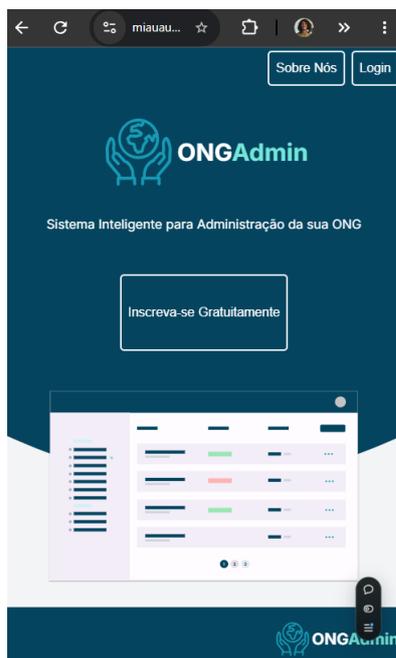


Fonte: BRITO, COHEN, PADULA 2024

## 4.2.2 Operação

Com a implementação dos ambientes de produção de cada módulo, o sistema é facilmente acessível através de um link, no qual o usuário é apresentado com a plataforma e opções de cadastro e autenticação:

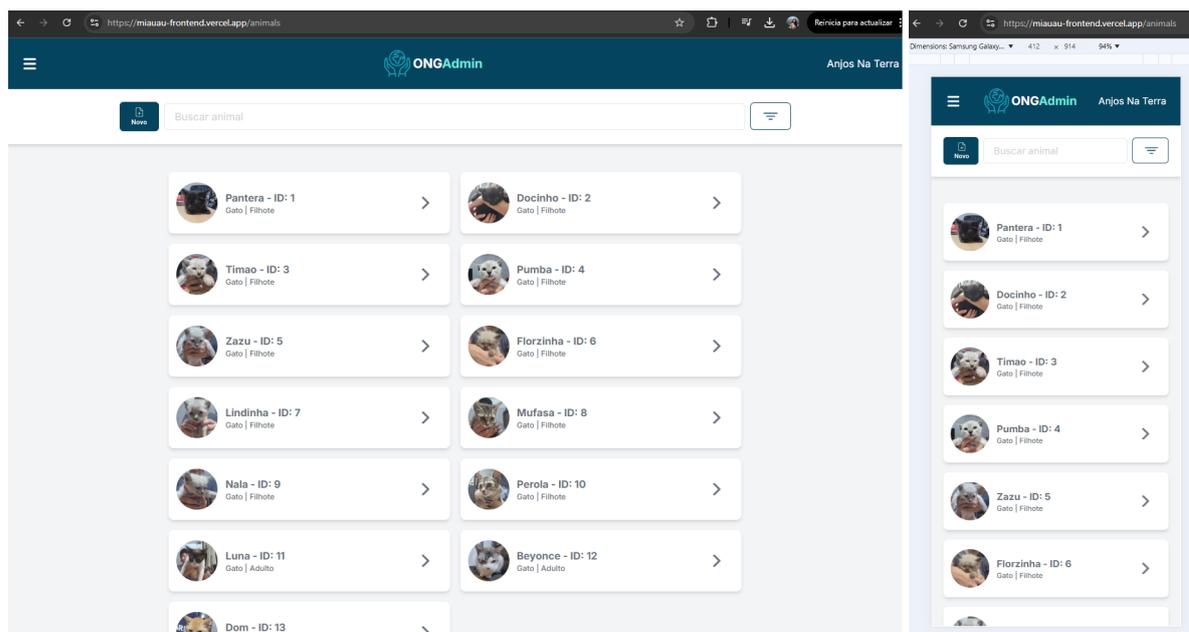
Figura 26 – Página Inicial do Sistema



Fonte: BRITO, COHEN, PADULA 2024

Após a autenticação, o usuário é direcionado para o dashboard de administração

Figura 27 – Módulo de Administração de Animais - Anjos Na Terra - Versão Desktop e Mobile



Fonte: BRITO, COHEN, PADULA 2024

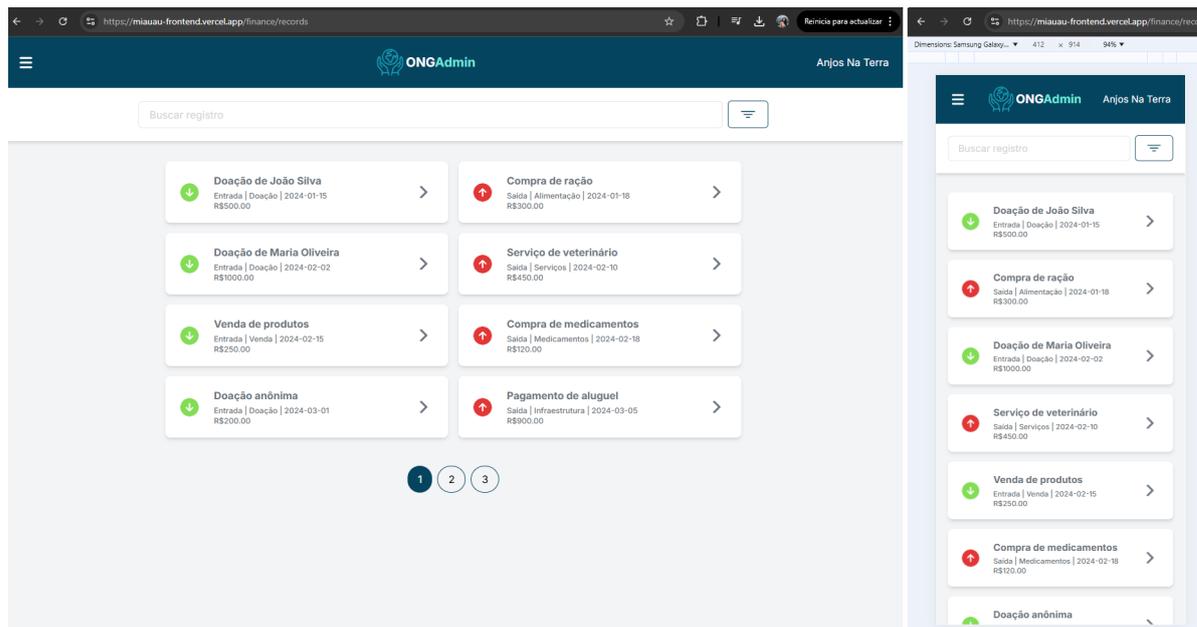
A navegação entre os módulos usados por essa ONG podem ser acessados na aba superior de navegação.

Figura 28 – Navegação entre Módulos - Anjos Na Terra



Fonte: BRITO, COHEN, PADULA 2024

Figura 29 – Módulo de Administração Financeira - Anjos Na Terra - Versão Desktop e Mobile



Fonte: BRITO, COHEN, PADULA 2024

### 4.3 Testes de Integração

Os testes da implementação foram realizados dentro de cada módulo em ambiente de desenvolvimento a fim de garantir sua funcionalidade e carregamento correto do ambiente de desenvolvimento para produção.

Com o sistema em operação foi possível fazer uma visita à ONG parceira para testes e colhimento do feedback. Foi feita a instrução aos voluntários da ONG de como interagir com o sistema, além do cadastro de alguns animais. Também foi colhido retorno de usabilidade, implementação e carga de uso.

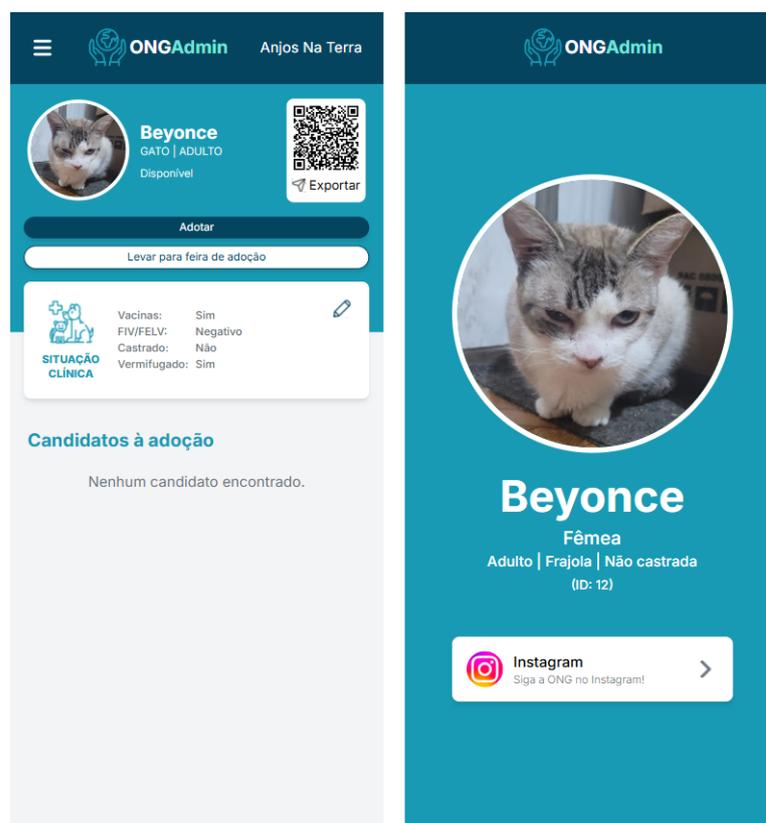
Figura 30 – Visita à ONG Anjos Na Terra



Fonte: BRITO, COHEN, PADULA 2024

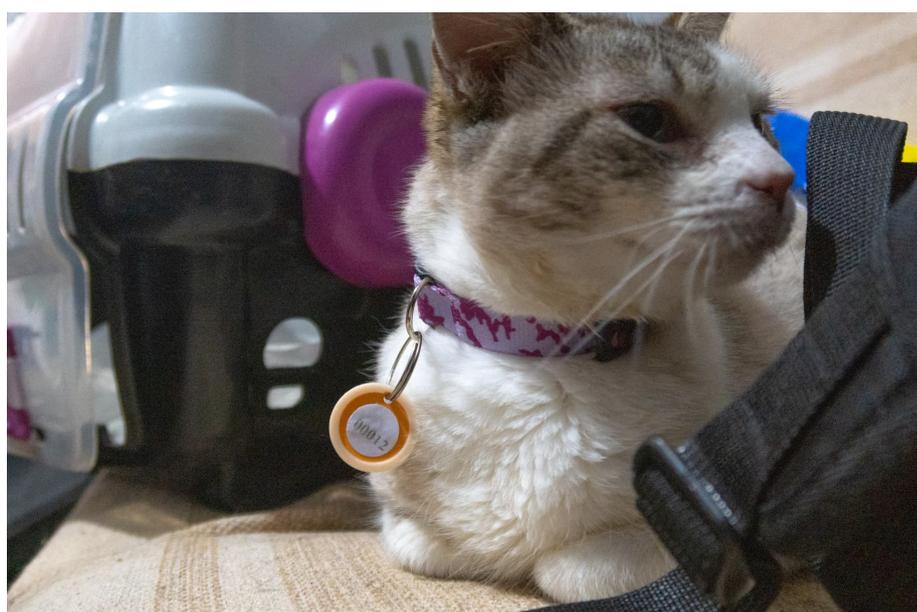
A visita foi providencial para uma melhor análise do contexto de organização e estrutura da ONG, reforçando a importância do caráter social do projeto e a viabilidade de uma atividade extensionista de parceria da Universidade com organizações do terceiro setor.

Figura 31 – Animal Cadastrado no Sistema - Visão administrativa e visão pública



Fonte: BRITO, COHEN, PADULA 2024

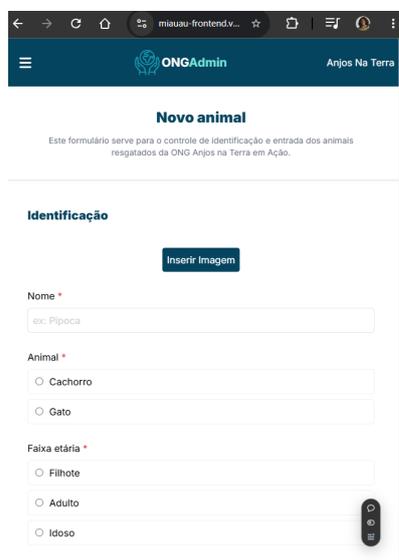
Figura 32 – Animal Cadastrado com coleira de Identificação



Fonte: BRITO, COHEN, PADULA 2024

Diversos animais foram nomeados, fotografados e identificados com coleira própria, seguido de um cadastro completo das condições de saúde, idade, castração e vermifugação.

Figura 33 – Formulário de Cadastro do Animal



The screenshot shows a mobile web interface for 'ONGAdmin' with the title 'Novo animal'. The page includes a header with the logo and 'Anjos Na Terra'. Below the title, there is a sub-header 'Identificação' and a button 'Inserir imagem'. The form contains several input fields: 'Nome \*' with a placeholder 'ex: Pipoca', 'Animal \*' with radio buttons for 'Cachorro' and 'Gato', and 'Faixa etária \*' with radio buttons for 'Filhote', 'Adulto', and 'Idoso'. A vertical scrollbar is visible on the right side of the form.

Fonte: BRITO, COHEN, PADULA 2024

Figura 34 – Formulário de Adoção



The screenshot shows a mobile web interface for 'ONGAdmin' with the title 'Informações pessoais'. The page includes a header with the logo and 'Anjos Na Terra'. Below the title, there is a sub-header 'Identificação' and a button 'Inserir imagem'. The form contains several input fields: 'Nome completo \*' with the value 'Ana da Silva', 'CPF \*' with a placeholder 'XXXX-XXXX-XX' and the instruction 'Insira somente números.', 'RG \*' with a placeholder 'XXXXXXXX-X' and the instruction 'Insira somente números.', 'Data de Nascimento \*' with a placeholder 'mm/dd/yyyy' and the instruction 'Necessário ter ao menos 21 anos.', and 'Celular \*' with a placeholder 'XXXXXXXX-XXXX' and the instruction 'Insira somente números.'. A vertical scrollbar is visible on the right side of the form.

Fonte: BRITO, COHEN, PADULA 2024

Também foi feita uma simulação do preenchimento dos formulários de candidatura à adoção para testes de fluxo e usabilidade, além do colhimento de feedback sobre o sistema de cadastro de animais para reformulação.

## 5 Considerações Finais

Nesta seção serão apresentadas as conclusões do texto, as contribuições do grupo e do projeto, e por fim perspectivas de continuidade do trabalho.

### 5.1 Conclusões do Projeto de Formatura

Em suma, notou-se um impacto muito positivo na criação do sistema de apoio e gerenciamento na instância de resgate e adoção de animais. O sistema foi capaz de contribuir na estruturação e organização dentro da ONG parceira, além de criar uma base para expansão do projeto para outros contextos. A transformação digital da organização mostrou-se eficaz, principalmente pela definição de estruturas e modelos a um contexto caótico. Essa estruturação não apenas aliviou a sobrecarga operacional da fundadora, mas também ampliou a capacidade da ONG de alcançar mais pessoas, facilitando processos como adoções e arrecadação de recursos.

Pode-se concluir que A transformação digital possibilita que ONGs, como a Anjos na Terra, maximizem seu impacto social ao automatizar tarefas operacionais, reduzir ineficiências e melhorar a gestão de recursos. Além disso, promove maior transparência e organização, fatores essenciais para atrair doadores e parceiros estratégicos. A plataforma digital é essencial por servir como um sistema centralizado que conecta as diversas operações da ONG, eliminando redundâncias e possibilitando a padronização de processos. Esse sistema também torna os dados mais acessíveis e utilizáveis para tomadas de decisão estratégicas.

Não obstante, os conceitos estudados e aplicados de Engenharia de Software foram fundamentais nesse processo. A definição de uma arquitetura robusta permitiu que a plataforma se tornasse escalável e modular, atendendo às necessidades imediatas da ONG e possibilitando sua expansão para outros cenários. Esse projeto demonstrou como soluções digitais podem transformar a gestão de projetos sociais, trazendo benefícios diretos às organizações e à sociedade. A escolha das tecnologias foi guiada pelos requisitos funcionais e não funcionais do projeto, como facilidade de integração, custo-benefício e compatibilidade com os recursos disponíveis. Critérios como comunidade ativa, suporte técnico e curva de aprendizado também foram considerados, assegurando que a tecnologia escolhida possa ser mantida e expandida por diferentes equipes no futuro.

A Plataforma Digital implementada foi capaz de atender às demandas, requisitos e necessidades do modelo de negócio social empregado, e a estrutura do projeto torna-se facilmente replicável em outros contextos do terceiro setor com a arquitetura definida. O

processo baseado em arquitetura permite lidar de forma mais eficiente com os desafios de distribuição e escalabilidade. Técnicas clássicas tendem a ser menos adaptáveis a cenários que demandam sistemas modulares e integrados. A abordagem arquitetural garante flexibilidade, manutenção mais simples e escalabilidade, características cruciais para um projeto que precisa atender diferentes contextos e evoluir com o tempo.

## 5.2 Contribuições

O principal propósito do trabalho foi atingido, trazendo uma contribuição positiva na sociedade através do projeto da universidade, ao mesmo tempo em que coloca a equipe em uma situação enriquecedora de aplicação dos conceitos estudados no curso. A equipe teve de sua autoria a modelagem, especificação, arquitetura e implementação da plataforma digital empregada, cujo software está disponível em código aberto. Os autores foram responsáveis pela instalação e instrução do uso do sistema dentro da ONG Anjos na Terra, auxiliando nos processos iniciais de cadastro dos usuários e cadastro dos animais.

## 5.3 Perspectivas de Continuidade

Por fim, um dos objetivos essenciais do projeto é a contribuição do projeto para o desenvolvimento das Atividades Extensionistas da nova grade curricular da Escola Politécnica da USP, nas quais os alunos poderão optar por dar continuidade no trabalho proposto. Os próximos passos de pesquisa e desenvolvimento da plataforma incluem:

- Expansão da instância de resgate de animais para ONGs em outros contextos;
- Integração dos sistemas de diversas ONGs para finalidades de melhor gestão de recursos e análise de dados pela prefeitura;
- Aplicação de sistemas de inteligência artificial em ferramentas de análise preditiva e recomendação;
- Criação de um modelo de busca em estilo 'Marketplace' para aumento da visibilidade e encontro de candidatos à adoção de animais;
- Ampliação do sistema para integração com empresas e plataformas interessadas no projeto social para aumento da captação de recursos.

Por conseguinte, o projeto cumpriu seu papel de atuar como base para que os futuros alunos da Universidade possam contribuir para a sociedade ao mesmo tempo em que desenvolvem empatia, visão de resolução de problemas e habilidades técnicas em sua formação como Engenheiros.

## Referências

ALBUQUERQUE, A. C. C. de. *Terceiro setor*. [S.l.]: Summus Editorial, 2006. Citado na página 14.

BECK, K. et al. *Manifesto for Agile Software Development*. 2001. Disponível em: <<https://agilemanifesto.org/>>. Acesso em: 03 nov. 2024. Citado na página 19.

CGI. *TIC ORGANIZAÇÕES SEM FINS LUCRATIVOS, Pesquisa Sobre o Uso das Tecnologias de Informação e Comunicação nas Organizações Sem Fins Lucrativos Brasileiras*. 2022. Disponível em: <<https://mapaosc.ipea.gov.br/arquivos/posts/3188-2023tic.pdf>>. Acesso em: 02 nov. 2024. Citado na página 15.

ECONOMATO. *Desafios na Gestão de ONGs: 6 Obstáculos e Como Vencê-los*. 2024. Disponível em: <<https://economato.com.br/desafios-na-gestao-de-ongs/>>. Citado na página 7.

FRANÇA, G. Como a tecnologia pode ajudar as ongs. *Portal do Impacto*, 2021. Disponível em: <<https://www.portaldoimpacto.com/como-a-tecnologia-pode-ajudar-as-ongs>>. Citado 2 vezes nas páginas 11 e 12.

GIFE. *A cada 10 ONGs abertas no Brasil, 3 encerraram suas atividades, indica estudo*. 2024. Disponível em: <<https://gife.org.br/a-cada-10-ongs-abertas-no-brasil-3-encerraram-suas-atividades-indica-estudo/>>. Acesso em: 07 abr. 2024. Citado na página 7.

GONG, C.; RIBIERE, V. Developing a unified definition of digital transformation. *Technovation*, v. 102, p. 102217, 2021. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0166497220300894>>. Citado na página 16.

IPB. *Número de animais de estimação em situação de vulnerabilidade mais do que dobra em dois anos, aponta pesquisa do IPB*. 2024. Disponível em: <<https://institutopetbrasil.com/fique-por-dentro/numero-de-animais-de-estimacao-em-situacao-de-vulnerabilidade-mais-do-que-dobra-em-dois-anos-ap>>. Acesso em: 08 abr. 2024. Citado na página 7.

IPEA. *Dinâmicas do terceiro setor no Brasil: Trajetórias de criação e fechamento de Organizações da Sociedade Civil (OSCs) de 1901 a 2020*. 2020. Disponível em: <<https://mapaosc.ipea.gov.br/arquivos/posts/3653-tddinamicaterceirosetorpublicacaopreliminar.pdf>>. Acesso em: 07 abr. 2024. Citado na página 7.

IPEA. *A IMPORTÂNCIA DO TERCEIRO SETOR PARA O PIB NO BRASIL E EM SUAS REGIÕES*. 2023. Disponível em: <<https://mapaosc.ipea.gov.br/arquivos/posts/9775-miolo-terceiro-setor-completo.pdf>>. Acesso em: 02 nov. 2024. Citado na página 14.

IPEA. *Mapa das Organizações da Sociedade Civil*. 2024. Disponível em: <<https://mapaosc.ipea.gov.br/mapa>>. Acesso em: 07 abr. 2024. Citado na página 7.

LAMARRE SHITAL CHHEDA, M. R. V. G. e. A. N. E. *The Value of Digital Transformation*. 2023. Disponível em: <<https://hbr.org/2023/07/the-value-of-digital-transformation>>. Acesso em: 1 nov. 2024. Citado na página 16.

- MEIRELLES, F. F. de S. et al. Impactos da pandemia da covid-19 no processo de transformação digital das ongs brasileiras. *Contemporary Journal*, v. 3, n. 8, p. 15–20, 2023. Disponível em: <[https://eaesp.fgv.br/sites/eaesp.fgv.br/files/u949/fernandodesouzameirelles\\_impactos\\_da\\_pandemia\\_da\\_covid-19\\_no\\_processo\\_de\\_transformacao\\_digital\\_das\\_ongs\\_brasileiras.pdf](https://eaesp.fgv.br/sites/eaesp.fgv.br/files/u949/fernandodesouzameirelles_impactos_da_pandemia_da_covid-19_no_processo_de_transformacao_digital_das_ongs_brasileiras.pdf)>. Citado na página 11.
- MELLO, C. B. Motivações na criação de um negócio social. *Universidade Federal do Rio Grande do Sul. Escola de Administração.*, 2016. Citado 2 vezes nas páginas 14 e 15.
- NIELSEN, J. *10 Usability Heuristics for User Interface Design*. 1994. Disponível em: <<https://www.nngroup.com/articles/ten-usability-heuristics/>>. Acesso em: 08 abr. 2024. Citado na página 44.
- ONU. *Sobre o nosso trabalho para alcançar os Objetivos de Desenvolvimento Sustentável no Brasil*. 2024. Disponível em: <<https://brasil.un.org/pt-br/sdgs>>. Acesso em: 02 nov. 2024. Citado 2 vezes nas páginas 15 e 16.
- PETRINI, M.; SCHERER, P.; BACK, L. Modelo de negócios com impacto social. *Revista de Administração de Empresas, SciELO Brasil*, v. 56, p. 209–225, 2016. Citado na página 14.
- QUEVEDO, B. N. *Atomic design no contexto de desenvolvimento: um relato de experiência*. 2023. Disponível em: <<http://dspace.sti.ufcg.edu.br:8080/xmlui/handle/riufcg/29290/>>. Acesso em: 22 abr. 2024. Citado na página 44.
- RICHARDS, M. *Software architecture patterns*. [S.l.]: O’Reilly Media, Incorporated 1005 Gravenstein Highway North, Sebastopol, CA . . . , 2015. v. 4. Citado na página 35.
- SHIVAKUMAR, S. K.; SETHI, S. *Building Digital Experience Platforms - A Guide to Developing Next-Generation Enterprise Applications*. 1. ed. [S.l.]: Apress, 2019. Citado 6 vezes nas páginas 12, 17, 18, 31, 35 e 37.
- SOUZA, A. F. B. D.; FERREIRA, B. M.; CONTE, T. Aplicando design thinking em engenharia de software: um mapeamento sistemático. In: *Ibero-American Conference on Software Engineering: Experimental Software Engineering Latin America Workshop (CibSE-ESELAW)*. [S.l.: s.n.], 2017. p. 719–732. Citado na página 12.
- WHITE, S. A. Introduction to bpmn. *Ibm Cooperation*, v. 2, n. 0, p. 0, 2004. Citado na página 25.
- WIERINGA, R. J. *Design Science Methodology: for Information Systems and Software Engineering*. [S.l.]: Springer-Verlag Berlin Heidelberg, 2014. Citado na página 13.