



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
CAMPUS TERESINA CENTRAL
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

EMANUEL VÍTOR DE SOUZA SILVA

PROJETO REFUSA: meta 3

TERESINA, PIAUÍ
2025

EMANUEL VÍTOR DE SOUZA SILVA

PROJETO REFUSA: META 3

Relatório Técnico de Desenvolvimento de Software apresentado para a conclusão da Disciplina de Elaboração de Projeto de Pesquisa do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí.

Orientador: Prof. Me . Rogério da Silva

TERESINA, PIAUÍ

2025

EMANUEL VÍTOR DE SOUZA SILVA

PROJETO REFUSA: meta 3

Relatório Técnico de Desenvolvimento de Software apresentado para a conclusão da Disciplina de Elaboração de Projeto de Pesquisa do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí.

Orientador: Prof. Me . Rogério da Silva

Aprovada em: 17 / 12 / 2025 .

BANCA EXAMINADORA:

Prof. Me. Rogério da Silva(Orientador)

Instituto Federal do Piauí (IFPI)

Prof. Me. Jose Ritomar Carneiro Torquato

Instituto Federal do Piauí (IFPI)

Prof. Me. Duany Dreyton Bezerra Sousa

Instituto Federal do Piauí (IFPI)

AGRADECIMENTOS

Agradeço a Deus pela vida e força para concluir esta jornada.

À memória de minha mãe, Janete, minha maior inspiração, que me apoiou incondicionalmente na escolha deste curso e durante toda a minha trajetória.

À minha esposa, Gabriela, e ao meu filho, Murilo, pelo amor, pela paciência durante as horas de ausência dedicadas aos estudos e por serem meu porto seguro.

Ao meu orientador, Prof. Me. Rogério da Silva, e aos professores do IFPI, pelo conhecimento compartilhado, e aos meus familiares pelo incentivo constante.

Aos meus familiares, agradeço pelo apoio constante, pelo incentivo e por terem sido a base fundamental para a minha trajetória.

A todos vocês, manifesto meus mais sinceros agradecimentos.

RESUMO

Atualmente, um dos problema que vem assolando a sociedade brasileira é a má distribuição de terras, onde agricultores familiares, que produzem a maior parte dos alimentos brasileiros, são os mais prejudicados. Para tentar resolver esse problema, o governo brasileiro criou o Programa Nacional de Crédito Fundiário, esse programa visa ajudar os agricultores familiares que atendem certos requisitos de renda. Mas ao liberar crédito, faz-se necessária uma fiscalização rigorosa de como esse crédito e a propriedade adquirida estão sendo utilizados. Com o avanço da tecnologia atual, um *software* de fiscalizações seria o ideal para resolver esse problema. Dando tanto uma velocidade na comunicação entre o fiscal e os gestores quanto uma facilidade no armazenamento dos dados das fiscalizações.

Palavras-chave: fiscalização, gestão, agricultura familiar, PNCF

ABSTRACT

Currently, one of the major challenges facing Brazilian society concerns the unequal distribution of land, a condition in which family farmers—who are responsible for producing the majority of Brazil's food supply—are disproportionately disadvantaged. In an attempt to mitigate this issue, the Brazilian government established the National Land Credit Program, designed to provide financial support to family farmers who meet specific income-related criteria. Nevertheless, the allocation of public credit necessarily requires stringent monitoring to ensure the proper use of both the financial resources and the land acquired through the program. Considering the advancements in contemporary information technology, the development of a specialized inspection software system emerges as an optimal solution, as it would enhance the efficiency of communication between inspectors and administrators while also improving the reliability and organization of inspection data management.

keywords: supervision, management, family farming, PNCF

LISTA DE ILUSTRAÇÕES

Figura 1 - Estrutura fundiária no Brasil.....	10
Figura 2 - Capa do jornal o Globo sobre o massacre de Eldorado dos Carajás no Pará.....	11
Figura 3 - Exemplo de arquitetura Offline First.....	20
Figura 4 - Visão geral do funcionamento integral do software.....	26
Figura 5 - Diagrama de classe do funcionamento do sistema.....	28
Figura 6 - Quadro de atividades.....	31
Figura 7 - Visualização Semanal.....	31
Figura 8 - Funcionamento do typescript.....	35
Figura 9 - Comparativo entre consultas REST e GraphQL.....	37
Figura 10 - Fluxos de fiscalização.....	46
Figura 11 - Tela inicial da aplicação móvel.....	54
Figura 12 - Tela principal de uma assentamento.....	55
Figura 13 - Aba de operação de financiamento.....	56
Figura 14 - Aba de dados do beneficiário.....	57
Figura 15 - Aba de tipificação e ocupação irregular.....	58
Figura 16 - Aba de infraestrutura do lote.....	59
Figura 17 - Painel principal da aplicação web com a barra lateral aberta.....	61
Figura 18 - Painel principal da aplicação web com a barra lateral aberta.....	61
Figura 19 - Listagem de Usuários.....	62
Figura 20 - Listagem de UTEs.....	63
Figura 21 - Listagem de Assentamentos.....	64
Figura 22 - Aba de parcelas de um assentamento.....	65
Figura 23 - Aba de problemas de um assentamento.....	66
Figura 24 - Aba de fiscalizações de um assentamento.....	66
Figura 25 - Tela iniciar uma fiscalização.....	68
Figura 26 - Tela iniciar uma fiscalização.....	69
Figura 27 - Barra lateral da aplicação móvel.....	71
Figura 28 - Componentes de sincronização de fiscalizações.....	72
Figura 29 - Tela inicial do graphql playground.....	73
Figura 30 - Código graphql que define o tipo usuário.....	74
Figura 31 - Código graphql que define o tipo assentamento.....	75
Figura 32 - Código graphql para busca de assentamentos.....	76
Figura 33 - Código graphql para criação de usuários.....	76

LISTA DE TABELAS

Tabela 01 - Cadastrar Unidade Técnicas Estaduais (UTE).....	48
Tabela 02 - Incluir Agentes Fiscais em uma fiscalização.....	50
Tabela 03 - Criar fiscalização.....	51

LISTA DE ABREVIATURAS E SIGLAS

IFPI	Instituto Federal de Educação, Ciência e Tecnologia do Piauí
PNCF	Programa Nacional de Crédito Fundiário
UTE	Unidade Técnica Estadual
FTRA	Fundo de Terras e da Reforma Agrária
MAPA	Ministério da Agricultura, Pecuária e Abastecimento
UX	User Experience (Experiência do Usuário)
UI	User Interface (Interface do Usuário)
PDF	Portable Document Format
HTML	HyperText Markup Language

SUMÁRIO

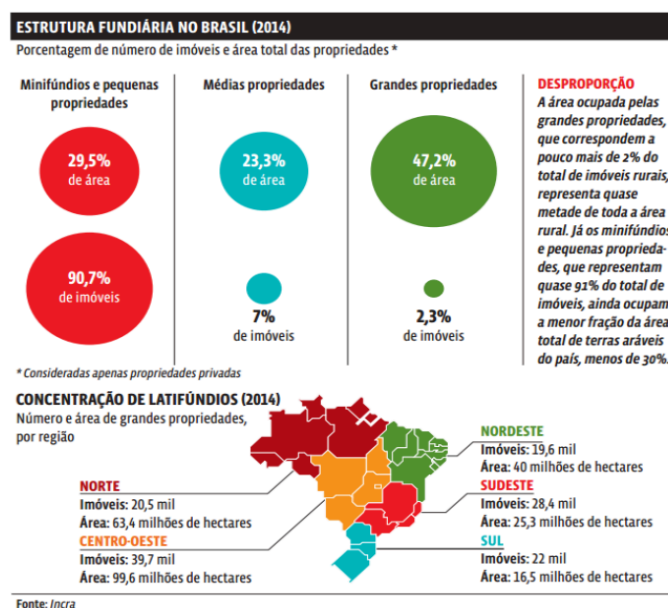
1 INTRODUÇÃO.....	10
1.1 PROGRAMA NACIONAL DE CRÉDITO FUNDIÁRIO.....	12
1.2 JUSTIFICATIVA.....	14
1.3 OBJETIVOS.....	15
1.4 ESTRUTURA DO TRABALHO.....	16
2 FUNDAMENTAÇÃO TEÓRICA.....	17
2.1 PROCESSOS DE DESENVOLVIMENTO DE SOFTWARE EM APLICAÇÕES WEB E MOBILE.....	18
2.2 METODOLOGIAS ÁGEIS.....	18
2.3 ARQUITETURA OFFLINE FIRST E PERSISTÊNCIA DE DADOS MÓVEIS.....	20
2.4 SEGURANÇA E PRIVACIDADE.....	23
3 METODOLOGIA.....	25
3.1 DIAGRAMAS.....	27
3.2 FERRAMENTAS DE DESENVOLVIMENTO.....	29
3.3 BACKEND.....	34
3.4 FRONTEND.....	41
3.5 MOBILE.....	44
4 SISTEMA DE FISCALIZAÇÃO DO PROGRAMA NACIONAL DE CRÉDITO FUNDIÁRIO.....	46
4.1 FLUXOS DE FISCALIZAÇÃO.....	46
4.2 REQUISITOS.....	48
4.3 APLICATIVO MÓVEL DE FISCALIZAÇÃO.....	53
4.4 INTERFACE DA PLATAFORMA WEB.....	61
4.5 OPERABILIDADE OFFLINE.....	70
4.6 BACKEND.....	73
4.7 CONSIDERAÇÕES FINAIS.....	78
5 CONCLUSÃO.....	79
REFERÊNCIAS.....	80
APÊNDICE A - LEVANTAMENTO DE REQUISITOS.....	83
ANEXO A - INTERFACES DO PROJETO.....	100
ANEXO B - FORMULÁRIO DE FISCALIZAÇÃO.....	113

1 INTRODUÇÃO

No contexto brasileiro contemporâneo, diversos problemas sociais persistem e impactam significativamente o desenvolvimento nacional. Dentre eles, destaca-se um que remonta ao período colonial: a desigualdade na distribuição de terras. Embora o Brasil seja o quinto maior país do mundo em extensão territorial, com uma área de 8.510.345,540 km² (IBGE, 2021), a concentração fundiária ainda é uma realidade marcante. Grandes latifúndios — extensas propriedades rurais pertencentes a um único proprietário — permanecem sob o controle de uma pequena parcela da população, enquanto agricultores familiares enfrentam, de forma crescente, a perda de suas terras e de suas condições de subsistência.

Tal cenário é particularmente preocupante, considerando que, em 2017, cerca de 70% dos alimentos consumidos no Brasil tiveram origem na agricultura familiar (PONTES, 2017). Em contrapartida, os grandes proprietários rurais destinam quase a totalidade de suas safras ao mercado externo, priorizando a exportação por razões econômicas, visto que as transações são realizadas em moeda estrangeira e contribuem para o equilíbrio da balança comercial do país. Ainda que essa dinâmica favoreça indicadores macroeconômicos, ela reforça desigualdades históricas e fragiliza a segurança alimentar nacional, evidenciando a urgência de políticas públicas voltadas à redistribuição de terras e à valorização da produção agrícola de base familiar.

Figura 1 - Estrutura fundiária no Brasil



Fonte: Blog Guia do estudante, 2016¹

Além das implicações econômicas e sociais decorrentes da concentração fundiária — como o fato de grande parte dos alimentos consumidos no Brasil ter origem na agricultura familiar —, essa desigualdade na posse de terras também tem gerado conflitos agrários de natureza violenta. Tais disputas resultam, com frequência, em confrontos armados, violações de direitos humanos e na morte de inúmeros trabalhadores rurais. Um dos episódios mais emblemáticos dessa realidade foi o massacre de Eldorado dos Carajás, ocorrido no estado do Pará, em 1996, quando 21 trabalhadores sem terra foram mortos, muitos deles com sinais de execução, durante uma operação conduzida pela Polícia Militar (CPT, 1997). Há ainda relatos de desaparecidos jamais localizados, o que evidencia a gravidade e persistência da violência no campo brasileiro.

¹ Disponível em:

<<https://guiadoestudante.abril.com.br/curso-enem-play/atualidades-brasil-educacao/>>
Acesso em abr. 2022.

Figura 2 - Capa do jornal o Globo sobre o massacre de Eldorado dos Carajás no Pará



Fonte: Acervo O Globo, 1996²

1.1 PROGRAMA NACIONAL DE CRÉDITO FUNDIÁRIO

Com o intuito de reduzir a desigualdade na distribuição de terras e mitigar os conflitos agrários decorrentes dessa concentração, o governo brasileiro instituiu o PNCF – Terra Brasil. Esse programa tem como objetivo principal viabilizar o acesso à terra por agricultores familiares e pequenos produtores rurais, permitindo-lhes adquirir suas próprias propriedades e realizar investimentos em infraestrutura produtiva e social. O financiamento é concedido com recursos do FTRA, garantindo condições de crédito diferenciadas e adaptadas à realidade econômica do público beneficiário.

Para ingressar no programa, o requerente deve entrar em contato com a UTE do estado onde a propriedade pretendida se encontra. A UTE é a entidade responsável pela execução do PNCF no âmbito estadual e atua sob a coordenação do Departamento de Gestão do Crédito Fundiário, vinculado à Secretaria de Agricultura Familiar e Cooperativismo do MAPA.

² Disponível em:

<<https://acervo.oglobo.globo.com/em-destaque/eldorado-de-carajas-massacre-dos-sem-terra-que-chocou-brasil-o-mundo-19101901>> Acesso em abr. 2022.

Dentre as principais atribuições da UTE estão: apoiar a execução do programa em conformidade com os normativos vigentes, supervisionar a implementação dos projetos pelos beneficiários, e promover a integração do PNCF com outras políticas públicas de desenvolvimento rural (IDAF, 2020). Essa estrutura descentralizada busca garantir maior eficiência, transparência e acompanhamento técnico das ações realizadas em cada estado.

O crédito disponibilizado aos agricultores familiares apresenta condições financeiras acessíveis, com taxas de juros que variam entre 0,5% e 4% ao ano, conforme a renda e o patrimônio do beneficiário. Além disso, o programa oferece carência de 36 meses e um prazo de até 25 anos para o pagamento do financiamento, o que contribui para a sustentabilidade econômica das famílias rurais e para o fortalecimento da agricultura familiar como base da segurança alimentar e do desenvolvimento regional.

1.1.1 Formulário de fiscalização

Os formulários de fiscalização utilizados pelo PNCF são instrumentos oficiais que orientam e padronizam o trabalho dos fiscais durante as vistorias realizadas nos assentamentos e lotes financiados pelo programa. Eles reúnem um conjunto estruturado de perguntas e campos obrigatórios que permitem registrar informações essenciais sobre a utilização da terra, a conformidade das atividades produtivas, a situação das famílias beneficiárias e eventuais irregularidades observadas.

Sua principal função é garantir uniformidade, rigor técnico e clareza no processo de avaliação, assegurando que todas as fiscalizações sigam os mesmos critérios e produzam dados consistentes e comparáveis. Além disso, os formulários atuam como documentos oficiais de comprovação das verificações realizadas, contribuindo para a tomada de decisão, a regularidade das operações e o controle público dos recursos do programa.

Para referência e consulta detalhada, todos os formulários utilizados no processo de fiscalização encontram-se integralmente apresentados nos anexos deste documento, permitindo ao leitor conhecer sua estrutura completa e sua aplicação prática no contexto do PNCF.

1.2 JUSTIFICATIVA

A fiscalização da aplicação dos recursos públicos destinados ao PNCF constitui uma etapa essencial para garantir a transparência, a eficiência e a efetividade das políticas de reforma agrária. Diante da concessão de subsídios e financiamentos por meio do programa, torna-se imprescindível assegurar que os valores disponibilizados estejam sendo utilizados de forma adequada, tanto no processo de aquisição das propriedades quanto no uso produtivo e sustentável das terras pelos beneficiários.

Nesse contexto, observa-se que os métodos tradicionais de fiscalização, baseados em formulários impressos e registros físicos, apresentam limitações significativas. Além das dificuldades logísticas associadas ao armazenamento de grandes volumes de documentos, esse modelo demanda tempo e recursos humanos elevados, uma vez que as informações coletadas em campo precisam ser entregues pessoalmente às UTEs para posterior análise e validação. Tal prática compromete a agilidade, a confiabilidade e a rastreabilidade dos dados, podendo gerar atrasos e inconsistências nos processos de monitoramento.

Com a crescente inserção da tecnologia digital no cotidiano e nas práticas administrativas, torna-se evidente a necessidade de modernizar o processo de fiscalização fundiária. A adoção de um sistema informatizado representa uma solução eficaz para superar as limitações do modelo manual. Por meio de um software integrado, os dados das fiscalizações podem ser armazenados em nuvem, garantindo segurança, disponibilidade e acessibilidade às informações em tempo real. Essa abordagem possibilita o acompanhamento remoto das atividades de campo, além de proporcionar rapidez na transmissão dos dados e redução de custos operacionais, consolidando-se como uma ferramenta fundamental para o fortalecimento da gestão pública e da governança fundiária no Brasil.

Nesse sentido, a implementação de soluções digitais para o monitoramento e a fiscalização de políticas públicas está alinhada às diretrizes da transformação digital do Estado brasileiro, que busca promover eficiência, transparência e inovação nos serviços públicos (BRASIL, 2021; CASTELLS, 2003). Assim, o desenvolvimento de um sistema de fiscalização digital contribui diretamente para o aperfeiçoamento

da administração pública, ao integrar tecnologia e gestão, fortalecendo os princípios da accountability e da governança eletrônica.

1.3 OBJETIVOS

O software proposto neste trabalho tem como finalidade desenvolver e consolidar o módulo de fiscalizações do projeto REFUSA, correspondendo à Meta 3 estabelecida no escopo do projeto, que prevê a criação de uma plataforma web integrada a um aplicativo móvel, ambos com controle de acesso e mecanismos de autenticação, destinados ao gerenciamento, distribuição e coleta de dados das fiscalizações.

De forma mais ampla, o objetivo geral do sistema é otimizar o processo de fiscalização dos projetos financiados pelo PNCF – Terra Brasil, garantindo maior eficiência, rastreabilidade e transparência na gestão das informações provenientes do campo.

Essa meta está estruturada em duas etapas principais. A primeira consiste no desenvolvimento de uma interface web interativa, capaz de consumir e manipular informações sobre os projetos aprovados, além de gerar, distribuir e monitorar demandas de fiscalização, tanto amostrais quanto extraordinárias. Essa etapa representa o núcleo de gestão administrativa e operacional da plataforma, servindo de ponto central para o acompanhamento das atividades de campo.

A segunda etapa abrange o desenvolvimento de um aplicativo móvel destinado aos fiscais responsáveis pelas vistorias em campo. Esse aplicativo permitirá receber as demandas emitidas pela interface web, registrar dados diretamente in loco, e emitir relatórios de gestão de forma automatizada, integrando-se em tempo real com o sistema principal. Dessa forma, busca-se reduzir a burocracia, minimizar erros de registro e agilizar o fluxo de informações entre os fiscais e as UTEs.

1.3.1 Objetivo geral

Desenvolver uma solução integrada capaz de modernizar e otimizar o processo de fiscalização do PNCF, garantindo eficiência, organização e confiabilidade na coleta e gestão das informações.

1.3.2 Objetivos específicos

- Criar um aplicativo móvel que permita a realização de fiscalizações em campo, inclusive em modo offline.
- Desenvolver um painel web para o gerenciamento de fiscalizações, assentamentos, UTEs, usuários e atribuições.
- Implementar uma API robusta para integrar os módulos da aplicação e garantir comunicação segura entre eles.
- Disponibilizar funcionalidades para registro, consulta e acompanhamento das fiscalizações.
- Permitir a geração de relatórios e formulários a partir dos dados enviados pelos fiscais no modelo oficial do PNCF.
- Garantir usabilidade, segurança dos dados e suporte a diferentes perfis de usuários.

1.4 ESTRUTURA DO TRABALHO

O presente relatório técnico está organizado em cinco capítulos principais, estruturados da seguinte forma para facilitar a compreensão do desenvolvimento e da implementação da solução tecnológica proposta:

- Capítulo 1 – Introdução: Apresenta a contextualização do problema, abordando a desigualdade na estrutura fundiária brasileira e o papel do Programa Nacional de Crédito Fundiário (PNCF). Define a justificativa para a criação de um software de fiscalização, destacando as limitações dos métodos manuais atuais, e estabelece os objetivos gerais e específicos do projeto.
- Capítulo 2 – Fundamentação Teórica: Discorre sobre os conceitos e tecnologias que embasam o desenvolvimento do sistema. Aborda processos de desenvolvimento de software para Web e Mobile, a utilização de Metodologias Ágeis (com foco no Scrum) e a importância da arquitetura

Offline First para operações em campo. Também trata dos aspectos de segurança da informação e conformidade com a Lei Geral de Proteção de Dados (LGPD).

- Capítulo 3 – Metodologia: Detalha o processo de engenharia de software adotado, desde a análise de requisitos até a implementação. Descreve as ferramentas de gestão e design utilizadas (Linear, Git, Figma) e especifica a *stack* tecnológica escolhida para o Backend (TypeScript, NestJS, GraphQL, Prisma, Wasabi), Frontend (Next.js, ReactJS) e Mobile (Dart, Flutter).
- Capítulo 4 – Aplicativo e Plataforma Web do PNCF: Apresenta os resultados práticos do projeto. Exibe os fluxos de fiscalização, a definição dos requisitos funcionais e não funcionais, e demonstra as interfaces finais desenvolvidas para o aplicativo móvel e para o painel de gestão web. Inclui também detalhes sobre a operabilidade em modo offline e a estrutura do backend.
- Capítulo 5 – Conclusão: Encerra o documento com as considerações finais sobre o trabalho desenvolvido, avaliando o cumprimento dos objetivos propostos, os benefícios da solução para a gestão pública e as perspectivas futuras para o software.

2 FUNDAMENTAÇÃO TEÓRICA

O desenvolvimento do software aqui descrito é motivado por um problema social persistente no contexto brasileiro: a desigualdade na distribuição de terras. Apesar de o Brasil ter uma vasta extensão territorial, a concentração fundiária é marcante, com grandes latifúndios controlados por uma pequena parcela da população.

Essa concentração é preocupante, pois cerca de 70% dos alimentos consumidos no Brasil têm origem na agricultura familiar, enquanto grandes proprietários priorizam a exportação. A desigualdade na posse de terras também gera conflitos agrários violentos.

O PNCF – Terra Brasil foi instituído pelo governo federal para mitigar essa desigualdade, permitindo o acesso à terra por agricultores familiares e pequenos produtores rurais. No entanto, a concessão de crédito exige uma fiscalização

rigorosa para garantir que os recursos e as propriedades adquiridas estejam sendo usados de forma adequada e sustentável.

Os métodos tradicionais de fiscalização, baseados em formulários impressos e registros físicos, muitas das vezes são ineficientes, demandam muito tempo e recursos, além de comprometerem a agilidade, a confiabilidade e a integridade dos dados coletados em campo. A adoção de um sistema informatizado é a solução para superar essas limitações, oferecendo armazenamento em nuvem, segurança, acessibilidade em tempo real, e agilidade na comunicação entre fiscais e gestores.

2.1 PROCESSOS DE DESENVOLVIMENTO DE *SOFTWARE* EM APLICAÇÕES *WEB* E *MOBILE*

O desenvolvimento de aplicações modernas — tanto *web* quanto *mobile* — exige processos de engenharia de *software* capazes de lidar com requisitos complexos, como responsividade, escalabilidade, segurança, interoperabilidade e disponibilidade contínua. Esses sistemas, frequentemente baseados em arquiteturas cliente-servidor, envolvem múltiplas camadas e interações constantes entre clientes (aplicações *web* ou *mobile*) e servidores responsáveis por processamento, armazenamento e entrega de dados. Conforme destaca (Sommerville, 2019), a multiplicidade de plataformas e dispositivos torna o processo de desenvolvimento mais desafiador, demandando abordagens que favoreçam flexibilidade e evolução incremental.

2.2 METODOLOGIAS ÁGEIS

Nesse cenário, as metodologias ágeis consolidaram-se como principal referência para gerenciamento e execução de projetos de software. Pressman e Maxim (2020) argumentam que métodos ágeis são especialmente adequados para o desenvolvimento de sistemas web e mobile devido ao caráter dinâmico desses ambientes, nos quais requisitos podem evoluir rapidamente com base no uso real, nas mudanças do negócio e no comportamento do usuário.

Entre as abordagens ágeis disponíveis, o *Scrum* destaca-se por sua estrutura simples, iterativa e adaptável. Segundo (Schwaber e Sutherland, 2020), o Scrum

organiza o trabalho em ciclos chamados sprints, que permitem a entrega contínua de incrementos funcionais do produto. Em aplicações web e mobile, esses incrementos podem abranger desde novos componentes de interface, ajustes de responsividade, melhorias em APIs *REST/GraphQL*, integrações com serviços externos, até otimizações de performance e recursos específicos de dispositivos (como GPS, câmera ou sensores).

O ciclo de vida do Scrum é composto por eventos fundamentais que permitem inspeção e adaptação contínuas:

- **Sprint:** Intervalo iterativo de 1 a 4 semanas no qual um incremento potencialmente utilizável é entregue. A duração reduzida dos ciclos favorece a rápida validação de funcionalidades em diferentes dispositivos e plataformas.
- **Sprint Planning:** Reunião dedicada a identificar e planejar os itens priorizados do Product Backlog, considerando requisitos de ambas as plataformas (web e mobile), capacidades técnicas e dependências.
- **Daily Scrum:** Reuniões diárias breves para alinhamento das atividades, identificação de impedimentos técnicos — como falhas de compilação, problemas de integração ou inconsistências entre front-end mobile e back-end — e coordenação da equipe.
- **Sprint Review:** Evento voltado à apresentação do incremento entregue, permitindo que stakeholders avaliem protótipos, interfaces responsivas, telas mobile, endpoints e funcionalidades gerais do sistema.
- **Sprint Retrospective:** Reunião focada na melhoria contínua, avaliando aspectos técnicos e organizacionais, como práticas de versionamento, testes automatizados, desempenho mobile, integração contínua (CI) e entrega contínua (CD).

O Scrum também define papéis essenciais:

- **Product Owner:** Responsável por maximizar o valor do produto, gerenciar o *Product Backlog* e garantir que os requisitos contemplem adequadamente as especificidades das plataformas web e mobile.

- *Scrum Master*: Facilita o processo, assegura a aplicação correta da metodologia e remove impedimentos que possam afetar o ritmo do desenvolvimento.
- *Development Team*: Equipe multifuncional responsável pela implementação técnica, incluindo desenvolvimento *frontend* (*web* e *mobile*), *backend*, testes, integração e preparação para deploy.

Os artefatos do Scrum proporcionam transparência e rastreabilidade ao processo:

- *Product Backlog*: Lista priorizada e continuamente atualizada que reúne funcionalidades, ajustes e requisitos técnicos. Em sistemas web e mobile, inclui histórias de usuário, componentes responsivos, recursos mobile nativos, endpoints e requisitos de infraestrutura.
- *Sprint Backlog*: Conjunto de itens selecionados para o sprint atual, acompanhado do plano de execução e dos critérios de aceitação, incluindo considerações específicas para as diferentes plataformas.

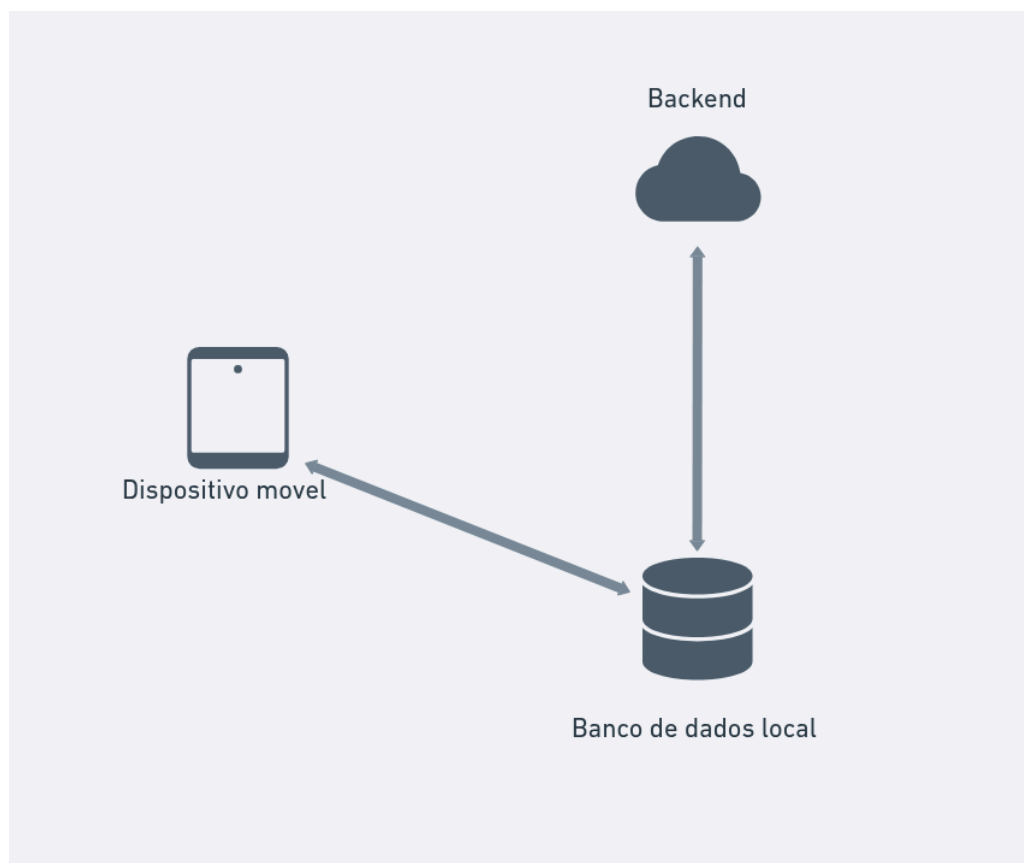
Assim, a utilização do Scrum no desenvolvimento de aplicações *web* e *mobile* oferece uma estrutura eficiente para lidar com a complexidade desses ambientes, garantindo flexibilidade, inspeção contínua, capacidade de adaptação e alinhamento constante entre equipe, usuários e objetivos estratégicos. Estudos na literatura reforçam que abordagens ágeis resultam em maior previsibilidade, redução de riscos e aumento da qualidade final do software (SOMMERVILLE, 2019; PRESSMAN; MAXIM, 2020; AHMAD; WASIM, 2023). Dessa forma, o Scrum apresenta-se como uma metodologia adequada e amplamente utilizada para conduzir projetos que envolvem a construção simultânea de aplicações web e mobile.

2.3 ARQUITETURA OFFLINE FIRST E PERSISTÊNCIA DE DADOS MÓVEIS

A natureza das atividades de fiscalização em áreas rurais, como os assentamentos do PNCF, impõe desafios significativos de conectividade. Para

mitigar a intermitência ou ausência de redes de dados, o projeto adota o paradigma *Offline First*. Nesta arquitetura, a aplicação móvel é projetada para tratar o armazenamento local como a "fonte de verdade" primária (source of truth), garantindo que todas as funcionalidades críticas operem independentemente de conexão com a internet.

Figura 3 - Exemplo de arquitetura Offline First



Fonte: Autoria Própria

2.3.1 Camada de persistência local

Para viabilizar o funcionamento desconectado, a aplicação utiliza um banco de dados embarcado no dispositivo. No ecossistema Mobile, uma das tecnologias mais utilizadas é o *SQLite*. Esse banco de dados é utilizado para dados estruturados e relacionais complexos, como os cadastros dos lotes e formulários de fiscalização, garantindo a integridade referencial das informações mesmo offline.

Essa camada atua como intermediária obrigatória. Toda interação do fiscal (criação de nota, foto ou preenchimento de formulário) é persistida imediatamente no banco local antes de qualquer tentativa de comunicação com a rede. Isso assegura que o fiscal não enfrente telas de carregamento ou falhas de requisição durante o trabalho de campo.

2.3.2 Estratégia de sincronização e gerenciamento de filas

A sincronização dos dados ocorre através de um padrão de "Fila de Sincronização" (*Sync Queue*). O funcionamento técnico segue o seguinte fluxo:

- **Intercepção:** Quando o fiscal salva uma fiscalização, o dado é gravado localmente e marcado com uma *flag* de estado (ex: *pending_sync: true*)
- **Sincronização de Dados:** A interface de gerenciamento do assentamento dispõe de uma funcionalidade dedicada ao envio de informações. Por meio de um botão de sincronização, o usuário pode iniciar manualmente a transmissão dos dados armazenados localmente para o servidor assim que houver conectividade.
- **Processamento em Lote:** O sistema lê os registros pendentes na fila local e os envia em pacotes (*batches*) para o servidor. Para arquivos pesados, como fotos e vídeos das vistorias, utiliza-se uma estratégia de transferência granular de arquivos, onde cada mídia (foto ou vídeo) é tratada como uma requisição HTTP independente. Diferente do envio em lote (*bulk upload*) — onde todos os arquivos são enviados em um único pacote — a abordagem unitária insere cada arquivo em uma fila de processamento assíncrona.

2.3.3 Resolução de Conflitos

Em cenários onde dados podem ser editados simultaneamente ou desatualizados, o sistema adota a estratégia de "Última Escrita Vence" (*Last Write Wins*) baseada em *timestamps*. Cada registro possui um carimbo de tempo da última modificação; ao sincronizar, o servidor compara o timestamp do dado

recebido com o do banco de dados central, garantindo a consistência final das informações.

2.4 SEGURANÇA E PRIVACIDADE

A segurança da informação é um pilar fundamental no desenvolvimento de sistemas de gestão pública como o PNCF, que lida com dados sensíveis de beneficiários e operações de crédito fundiário. Este capítulo detalha as estratégias de segurança e conformidade adotadas, com ênfase na Lei Geral de Proteção de Dados (LGPD) e nos mecanismos de fiscalização em campo para garantir a integridade do processo.

2.4.1 Conformidade com a lei geral de proteção de dados (LGPD)

A Lei nº 13.709/2018, ou LGPD, estabelece regras sobre a coleta, o uso, o tratamento e o armazenamento de dados pessoais. A conformidade com esta lei é mandatória para o PNCF, dada a natureza das informações manipuladas.

O sistema processa dados cadastrais de agricultores familiares, informações financeiras relacionadas ao crédito e dados de localização (georreferenciamento). Tais informações exigem níveis de proteção e acesso restrito para prevenir incidentes de segurança.

O tratamento dos dados deve se limitar ao propósito específico do projeto: a gestão e fiscalização do crédito fundiário. Qualquer coleta deve ser estritamente necessária para atingir essa finalidade.

A arquitetura do sistema deve empregar criptografia de dados e autenticação segura para proteger as informações contra acessos não autorizados. A segregação de funções por meio do controle de acesso baseado em papéis (RBAC) é crucial para garantir que apenas o Administrador, o Gestor de UTE e o Fiscal tenham acesso às informações estritamente necessárias para suas atribuições.

2.4.2 Georreferenciamento e Confirmação de Lote

A fiscalização *in loco* é a etapa mais crítica para a validação da aplicação do crédito fundiário. O *software* foi desenvolvido com o objetivo de estabelecer mecanismos para garantir que o fiscal realmente esteve no local e para assegurar a veracidade dos dados coletados. A identificação precisa da localização do fiscal é uma medida de segurança e rastreabilidade.

O aplicativo móvel utilizado pelo fiscal deve obrigatoriamente capturar e enviar as coordenadas geográficas (localização) no momento do registro da fiscalização.

O sistema deve cruzar a localização enviada com o lote ou a área de intervenção do PNCF que está sendo fiscalizado. O registro só deve ser permitido se a localização do fiscal estiver dentro de um raio de tolerância predefinido do lote associado àquela operação de crédito. O envio da localização garante que a fiscalização ocorreu *in loco*, conforme o objetivo do programa, e fornece aos gestores a rastreabilidade do processo.

2.4.3 Provas Visuais: Obrigatoriedade de Fotos e Vídeos

Para mitigar fraudes e fornecer evidências irrefutáveis sobre o estado da propriedade, o *software* exige a captura de multimídia no local da fiscalização. O fiscal deve anexar fotos e, em casos específicos, vídeos do local. Essas mídias servem como prova da existência e da situação do imóvel, bem como da aplicação dos recursos do crédito fundiário. O aplicativo deve ser configurado para:

- Captura em Tempo Real: Permitir a captura das fotos/vídeos apenas pela câmera do dispositivo no momento da fiscalização, impedindo o anexo de mídias pré-existentes da galeria.
- Metadados: Inserir automaticamente metadados na mídia, incluindo o carimbo de tempo (*timestamp*) e as coordenadas geográficas capturadas no momento exato do clique, reforçando a validade legal da prova.

Esses mecanismos combinados de controle de acesso, georreferenciamento e provas visuais tornam o projeto uma plataforma segura e confiável, essencial para a integridade do PNCF.

3 METODOLOGIA

O desenvolvimento de *software* para o projeto PNCF seguiu um processo estruturado que abrangeu todas as fases essenciais da engenharia de *software*, desde a concepção inicial até a implementação e futura manutenção de sua solução tecnológica. Este processo é fundamental para garantir que o produto final, que inclui a plataforma web e o aplicativo móvel, seja confiável, escalável e eficiente.

O processo de desenvolvimento consiste na definição, especificação, design, codificação, documentação, testes e manutenção de soluções tecnológicas. Para esse *software*, essa jornada seguiu uma série de etapas bem definidas:

1 - Análise e Definição de Requisitos (Identificação de Necessidades)

A fase inicial focou na identificação das necessidades dos usuários (Administrador, Gestor de UTE e Fiscal).

- Levantamento de Requisitos: Foram analisados documentos de fiscalização e reuniões com os representantes do Ministério do Desenvolvimento Agrário e Agricultura Familiar (MDA) para levantar os requisitos do software.
- Especificação: O resultado foram os Requisitos Funcionais (F-01 a F-12) e a descrição dos Casos de Uso (C-01 a C-12) , que detalham as funções necessárias, como o Cadastrar Usuários e o Realizar fiscalização em modo *online* ou *offline*.

2. Design do Sistema (Modelagem e Protótipos)

Esta fase envolveu a concepção da arquitetura e das interfaces, empregando boas práticas e princípios de engenharia de software.

- Arquitetura: Optou-se pela arquitetura cliente-servidor , essencial para a integração entre a interface web (gestão) e o aplicativo móvel (campo).

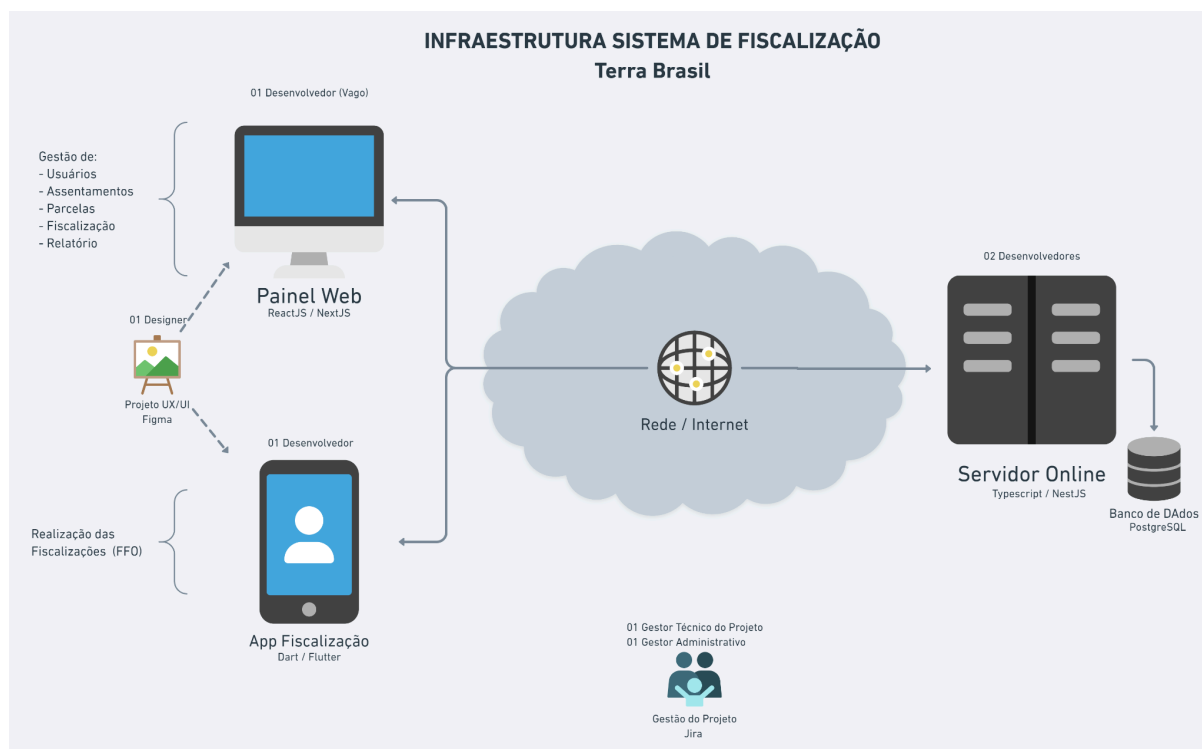
- Modelagem (UML): A estrutura do sistema foi demonstrada por meio de diagramas utilizando a linguagem *Unified Modeling Language* (UML) , como o Diagrama de Classes e o Fluxo de fiscalização.
- Design de Interface: O *software* Figma foi utilizado para desenvolver toda a concepção visual e a criação dos protótipos funcionais das interfaces *web* e *mobile*, garantindo a usabilidade e consistência visual.

3. Implementação e Codificação

A implementação é a fase de codificação, onde as especificações de design são transformadas em código funcional. Para este projeto, foi utilizada uma *stack* de tecnologias modernas:

- *Backend*: Desenvolvido em *TypeScript* e *NestJS*.
- *Frontend*: Construído com *Next.js* e *React.js*.
- *Mobile*: Utilizou *Flutter*, permitindo que a equipe se concentrasse na lógica de negócio e menos em configurações de infraestrutura.
- Controle de Versão: O *Git* e o *GitHub* foram cruciais para a colaboração simultânea e o rastreamento preciso de alterações no código-fonte.
- Design: Para os protótipos de tela foi utilizado a ferramenta *Figma*.

Figura 4 - Visão geral do funcionamento integral do software.



Fonte: Autoria Própria

3.1 DIAGRAMAS

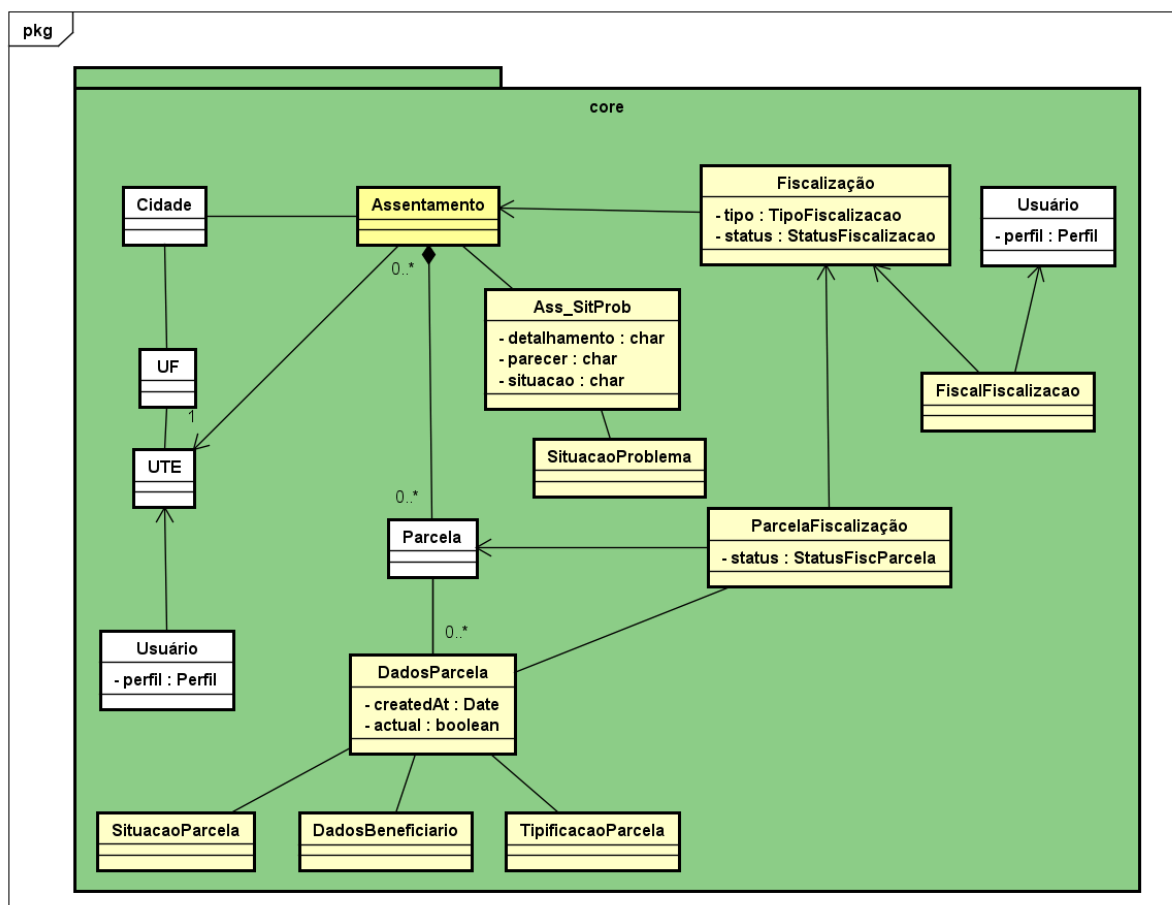
Com o objetivo de representar de forma clara e padronizada a estrutura do sistema em desenvolvimento, optou-se pela utilização do diagrama de classes da UML (Unified Modeling Language). A UML constitui uma linguagem amplamente reconhecida na engenharia de software por sua capacidade de descrever, de maneira visual e sistemática, os elementos estruturais e comportamentais de um sistema. O diagrama de classes, em particular, permite evidenciar as entidades envolvidas, seus atributos, métodos e os tipos de relacionamentos estabelecidos entre elas. Dessa forma, sua adoção contribui para uma melhor compreensão do domínio do problema, facilita a comunicação entre os membros da equipe de desenvolvimento e promove maior precisão na etapa de modelagem, reduzindo ambiguidades durante o processo de implementação.

3.1.2 Diagrama de Classes

O diagrama de classes constitui um dos elementos centrais da Unified Modeling Language (UML), sendo amplamente empregado na engenharia de software para representar a estrutura estática de sistemas orientados a objetos. De acordo com Booch, Rumbaugh e Jacobson (2005), criadores da UML, esse tipo de diagrama descreve as classes que compõem o sistema, seus atributos, operações e os diferentes tipos de relacionamentos estabelecidos entre elas, como associações, generalizações e composições. Tal representação possibilita uma compreensão clara e organizada do domínio do problema, permitindo identificar responsabilidades, dependências e a arquitetura lógica que guiará as etapas subsequentes de desenvolvimento.

Fowler (2004), destaca que o diagrama de classes desempenha papel fundamental ao reduzir ambiguidades durante a análise e o projeto, uma vez que fornece uma linguagem visual comum a todos os envolvidos no processo de desenvolvimento. Sua utilização favorece a comunicação entre analistas, projetistas e desenvolvedores, garantindo maior alinhamento conceitual e promovendo consistência estrutural ao longo de todo o ciclo de vida do software. Dessa forma, o diagrama de classes configura-se como um instrumento essencial para modelagem, documentação e validação de sistemas complexos, sendo amplamente recomendado em metodologias modernas de engenharia de software.

Figura 5 - Diagrama de classe do funcionamento do sistema



Fonte: Autoria Própria

3.2 FERRAMENTAS DE DESENVOLVIMENTO

Para o desenvolvimento deste *software*, optou-se pela arquitetura cliente-servidor, um modelo amplamente utilizado em sistemas distribuídos, no qual há um cliente — geralmente representado por um navegador de internet ou aplicativo móvel — que se conecta a um servidor, responsável pelo processamento das solicitações e pelo fornecimento de respostas. A comunicação entre cliente e servidor é realizada predominantemente por meio do Hypertext Transfer Protocol (HTTP), um protocolo da camada de aplicação, conforme definido no Modelo OSI.

O HTTP foi concebido para sistemas de informação hipermídia, distribuídos e colaborativos, permitindo que clientes requisitem recursos específicos ao servidor, que, por sua vez, processa essas requisições e retorna as respostas correspondentes (BERNERS-LEE; FIELDING; FRYSTYK, 1996). Essa arquitetura

apresenta vantagens significativas, incluindo facilidade de manutenção, escalabilidade do sistema e separação clara entre a interface do usuário e a lógica de processamento, características essenciais para aplicações web modernas e integradas a dispositivos móveis.

No contexto do projeto, a arquitetura cliente-servidor permite a integração entre a interface web e o aplicativo móvel, de modo que as demandas de fiscalização sejam geradas, distribuídas e monitoradas pela plataforma web, enquanto os fiscais podem registrar dados in loco por meio do aplicativo, com sincronização em tempo real com o servidor. Além disso, o servidor central gerencia o armazenamento dos dados na nuvem, garantindo segurança, disponibilidade e escalabilidade, enquanto o processamento das informações ocorre de forma assíncrona, permitindo que múltiplos arquivos de fiscalizações, fotos e vídeos sejam enviados, processados e integrados ao sistema sem comprometer o desempenho ou a experiência do usuário.

Dessa forma, a escolha da arquitetura cliente-servidor torna-se essencial para atender às necessidades funcionais e não funcionais do sistema, fornecendo uma estrutura robusta, flexível e confiável para o gerenciamento das fiscalizações, promovendo eficiência operacional, agilidade no processamento de dados e qualidade na entrega de informações para os gestores do programa.

3.2.1 Linear

O Linear é uma ferramenta voltada para o gerenciamento de projetos e controle de tarefas, desenvolvida especialmente para equipes de engenharia de software, design e produto. Assim como outras plataformas de gestão, o Linear oferece recursos para documentação, registro de anotações e acompanhamento do progresso do trabalho em quadros organizacionais, mas seu grande diferencial está na simplicidade, velocidade e integração com fluxos ágeis de desenvolvimento.

Essa ferramenta foi escolhida por combinar eficiência operacional com boas práticas do desenvolvimento ágil, oferecendo funcionalidades como criação e acompanhamento de sprints, gestão de *issues* (tarefas e bugs), planejamento de roadmaps e emissão de relatórios de desempenho. O Linear também incorpora visualizações de progresso, como gráficos de *burndown* e *burnup*, que auxiliam as

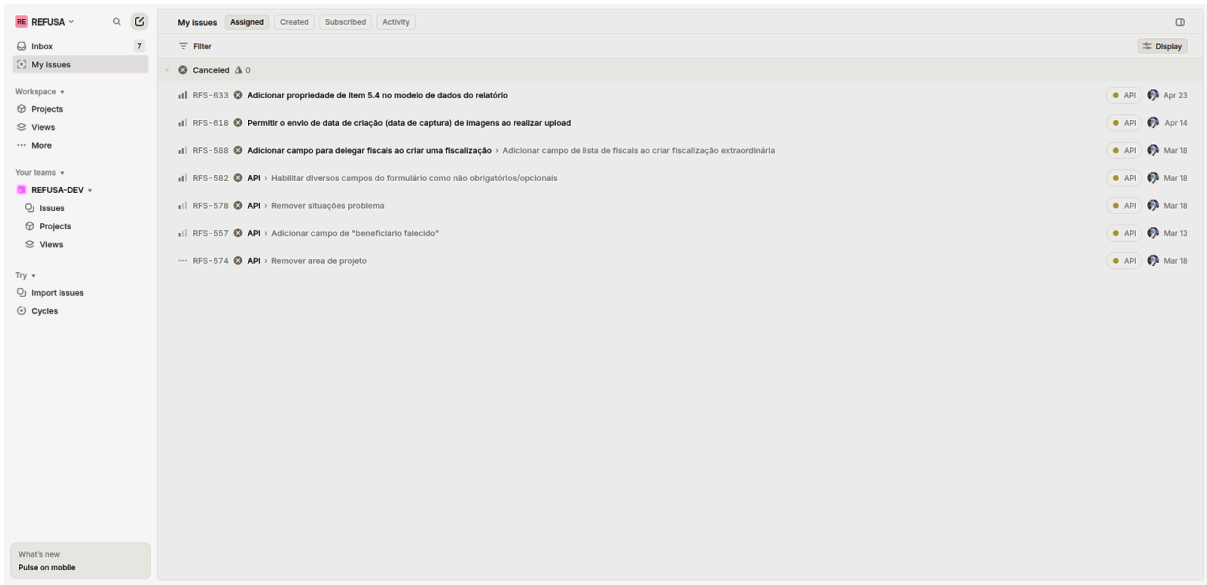
equipes a monitorar a evolução das entregas e a identificar gargalos no fluxo de trabalho.

Entre os principais benefícios da utilização do Linear destacam-se:

- Interface intuitiva e de alta performance, que reduz o tempo gasto na gestão e atualização das tarefas;
- Automatização de fluxos de trabalho, permitindo que o time se concentre mais no desenvolvimento e menos na administração de tarefas;
- Integração com diversas ferramentas do ecossistema de desenvolvimento, como GitHub, GitLab e Slack, o que facilita o acompanhamento contínuo das entregas;
- Transparência e colaboração aprimorada entre os membros da equipe, favorecendo a comunicação e a rastreabilidade das decisões de projeto;
- Análises de produtividade e métricas ágeis, que contribuem para uma melhoria contínua do processo de desenvolvimento.

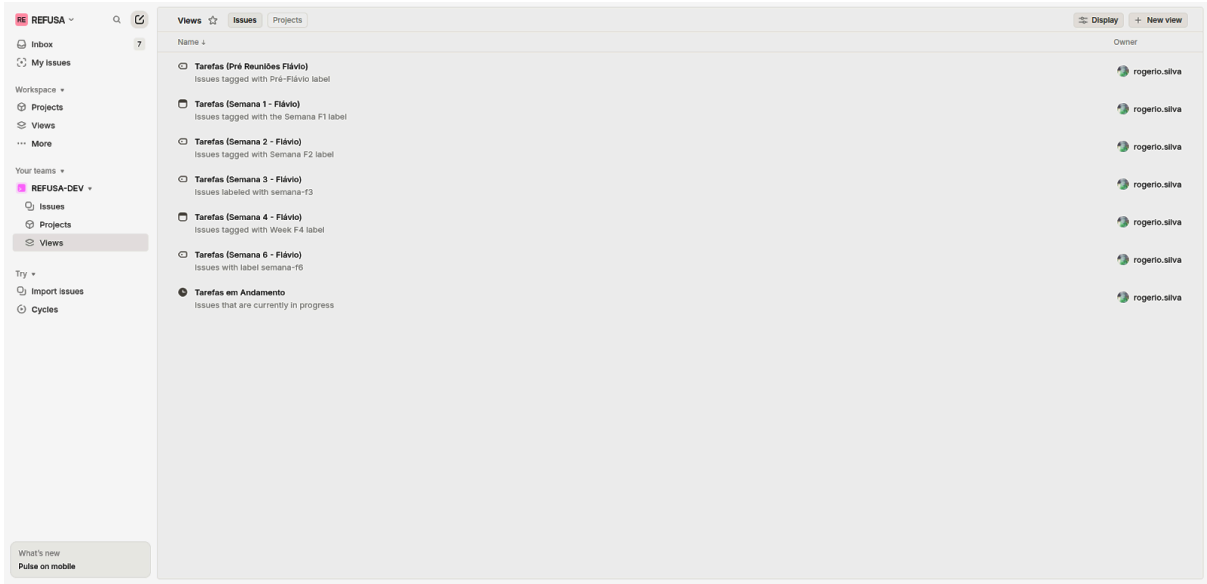
No contexto do software, a escolha do Linear se mostrou particularmente adequada por sua integração fluida com práticas do desenvolvimento ágil, metodologia adotada ao longo de toda a execução do projeto. A ferramenta possibilitou o planejamento, acompanhamento e controle das atividades de forma organizada, permitindo visibilidade total sobre o andamento das tarefas, monitoramento das metas estabelecidas e agilidade na resolução de impedimentos. Além disso, sua interface moderna e responsiva facilitou a colaboração entre os membros da equipe, mesmo em um ambiente de desenvolvimento distribuído, garantindo eficiência na comunicação e na gestão das entregas. Assim, o uso do Linear contribuiu significativamente para a organização, rastreabilidade e qualidade do processo de desenvolvimento do software, alinhando-se aos princípios da metodologia ágil e aos objetivos estratégicos do projeto.

Figura 6 - Quadro de atividades



Fonte: Autoria Própria

Figura 7 - Visualização Semanal



Fonte: Autoria Própria

3.2.2 Git e GitHub

O Git é um sistema de controle de versão distribuído (*Distributed Version Control System* – DVCS), de código aberto, desenvolvido por Linus Torvalds em 2005, com o objetivo de oferecer uma solução eficiente, segura e escalável para o gerenciamento de código-fonte de projetos de software. Essa ferramenta possibilita o rastreamento preciso de alterações, o controle de histórico de versões e a colaboração simultânea entre diversos desenvolvedores, características essenciais para equipes que trabalham de forma distribuída ou em projetos de grande porte.

A escolha pelo uso do Git fundamenta-se, principalmente, em sua arquitetura descentralizada, na qual cada desenvolvedor mantém uma cópia completa do repositório localmente, incluindo todo o histórico de versões. Esse modelo proporciona maior autonomia, reduz a dependência de conexões contínuas com servidores centrais e garante resiliência, uma vez que os dados do projeto permanecem replicados entre os colaboradores.

Além de suas vantagens técnicas, o Git consolidou-se como padrão de mercado no desenvolvimento de software moderno, sendo amplamente adotado por empresas de grande relevância global, como Google, Facebook e Netflix, bem como pela maioria das comunidades de software livre. Sua flexibilidade, velocidade de operação e ampla integração com plataformas de hospedagem e colaboração, como GitHub, GitLab e Bitbucket, tornam-no uma ferramenta essencial para a gestão eficiente do ciclo de vida do software, contribuindo diretamente para a qualidade, rastreabilidade e organização do processo de desenvolvimento.

3.2.3 Figma

O Figma é um *software* voltado para o design de interfaces e a prototipagem de projetos digitais, amplamente utilizado no desenvolvimento de aplicações *web* e móveis. Trata-se de um editor gráfico vetorial baseado na *web*, o que significa que todas as suas funcionalidades operam diretamente em um navegador, dispensando instalações complexas e facilitando o acesso remoto.

Um dos principais diferenciais do Figma em relação a outros *softwares* de design é sua capacidade de colaboração em tempo real. Essa funcionalidade

permite que vários profissionais trabalhem simultaneamente em um mesmo projeto, visualizando instantaneamente as alterações feitas por seus colegas e interagindo dentro do próprio ambiente de edição. Esse modelo colaborativo tem se tornado cada vez mais relevante em equipes multidisciplinares e distribuídas, especialmente em projetos que seguem metodologias ágeis de desenvolvimento (HARADA, 2022).

No contexto deste trabalho, toda a concepção visual do software proposto foi desenvolvida utilizando o Figma, abrangendo desde o planejamento da identidade visual até a criação dos protótipos funcionais das interfaces web e mobile. Essa etapa foi fundamental para garantir consistência visual, usabilidade e alinhamento entre a equipe de design e desenvolvimento. Os protótipos e interfaces resultantes estão disponibilizados na seção de anexos deste documento (ver Anexo 03), servindo como referência visual e funcional para a implementação do sistema descrito.

3.3 BACKEND

O *backend* corresponde à camada lógica e estrutural de uma aplicação, sendo responsável por gerenciar os dados, processar informações e implementar as regras de negócio que sustentam o funcionamento do sistema. Em outras palavras, é a parte que atua nos bastidores da aplicação, garantindo que todas as operações realizadas pelo usuário na interface (*frontend*) sejam devidamente tratadas, validadas e armazenadas.

Essa camada é composta por servidores, bancos de dados, APIs (*Application Programming Interfaces*) e serviços responsáveis pelo processamento e comunicação entre diferentes partes do sistema. O *backend* define como os dados são recebidos, manipulados e retornados ao cliente, assegurando aspectos essenciais como segurança, integridade, desempenho e escalabilidade da aplicação.

No contexto do desenvolvimento de sistemas modernos, o *backend* é essencial para garantir a confiabilidade das operações, uma vez que centraliza o controle das regras de negócio — isto é, os critérios e lógicas que determinam o comportamento do *software* de acordo com os objetivos do domínio da aplicação.

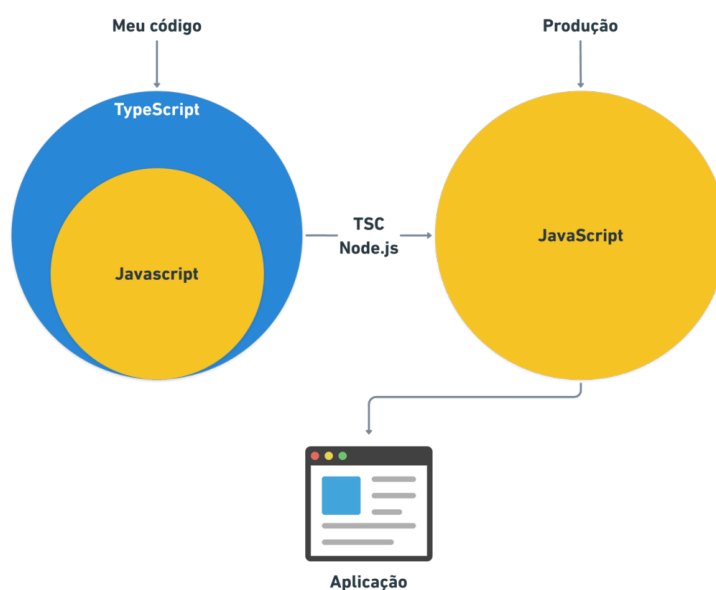
3.3.1 Linguagem

O *backend* do sistema desenvolvido neste projeto foi implementado utilizando a linguagem de programação *TypeScript*, escolhida por sua robustez, clareza sintática e ampla adoção no ecossistema do *Node.js*.

O *TypeScript* é uma linguagem criada pela *Microsoft* que se define como um superconjunto estrito de *JavaScript*, adicionando a ele tipagem estática opcional e recursos avançados de orientação a objetos. Essa característica permite descrever a estrutura e o comportamento dos objetos de forma explícita, favorecendo a documentação do código, a prevenção de erros em tempo de compilação e a manutenção de projetos de grande porte (TYPESCRIPT, 2021). Além disso, o *TypeScript* oferece melhor suporte a ferramentas de desenvolvimento (IDEs), com funcionalidades como autocompletar, inferência de tipos e refatoração assistida, o que aumenta significativamente a produtividade e a confiabilidade do código.

Para que o código escrito em *TypeScript* possa ser executado, ele é transpilado para *JavaScript*, a linguagem nativamente interpretada pelos navegadores e por plataformas baseadas no motor V8. No caso deste projeto, a execução do código *JavaScript* no servidor é viabilizada pelo *Node.js*, um ambiente de execução open source, multiplataforma, desenvolvido com base no motor V8 do Google Chrome, que possibilita a execução de código *JavaScript* fora do ambiente do navegador (NODE.JS, 2021).

O *Node.js* distingue-se por seu modelo de execução assíncrono e orientado a eventos, o qual permite elevada escalabilidade e desempenho superior em aplicações que demandam intenso processamento de operações de entrada e saída (I/O). Tal arquitetura mostra-se especialmente adequada para sistemas que lidam com um grande número de requisições simultâneas, interações contínuas com bancos de dados e comunicação frequente com serviços externos. Nesse contexto, a adoção conjunta de *TypeScript* e *Node.js* configura uma base tecnológica moderna e robusta para o desenvolvimento da camada de *backend* do sistema, pois combina tipagem estática, maior previsibilidade de comportamento, segurança na escrita do código e facilidade de manutenção. Essa integração contribui significativamente para a confiabilidade da aplicação e para a redução de erros durante o ciclo de desenvolvimento.

Figura 8 - Funcionamento do typescript

Fonte: Blog do Otávio Miranda, 2020³

3.3.2 NestJS

Para auxiliar no desenvolvimento da aplicação back-end, foi utilizado o framework *NestJS*, uma ferramenta moderna e de código aberto que se destaca por combinar produtividade, flexibilidade arquitetural e aderência a boas práticas de engenharia de *software*.

O *NestJS* foi escolhido principalmente por sua capacidade de abstrair tarefas complexas, permitindo ao desenvolvedor concentrar-se na lógica de negócio em vez de em configurações de infraestrutura. Além disso, o *framework* oferece uma arquitetura modular e altamente flexível, possibilitando a adoção tanto de sua estrutura nativa, baseada em módulos, controladores e serviços, quanto de outras arquiteturas personalizadas conforme a necessidade do projeto.

Outro fator determinante na escolha do *NestJS* foi o seu ecossistema rico e integrado, que oferece suporte nativo ou extensões para ferramentas amplamente

³ Disponível em:

<<https://www.otaviomiranda.com.br/2020/typescript-uma-longa-introducao/>> Acesso em abr. 2022.

utilizadas, como *TypeORM*, *Prisma*, *Passport*, *Swagger*, *GraphQL*, entre outras, tornando o processo de integração e manutenção mais ágil e padronizado.

Um dos pontos fortes do *framework* é o seu sistema de injeção de dependências, que segue o princípio da Inversão de Dependência (I) e o da Injeção de Dependência (D), representados pela sigla I e D do acrônimo SOLID. Esse mecanismo é implementado por meio de um container de injeção de dependências que utiliza o padrão de projeto Singleton — um padrão criacional que garante a existência de apenas uma instância global de cada classe durante a execução da aplicação. Essa abordagem contribui para um melhor controle do uso de memória, além de favorecer a testabilidade e a manutenção do código.

Dessa forma, o *NestJS* provê uma base sólida e organizada para o desenvolvimento da aplicação, conciliando performance, escalabilidade e boas práticas arquiteturais, o que o torna uma escolha adequada para sistemas modernos e de alta complexidade.

3.3.2 Banco de dados e ORM - *Object Relational Mapping*

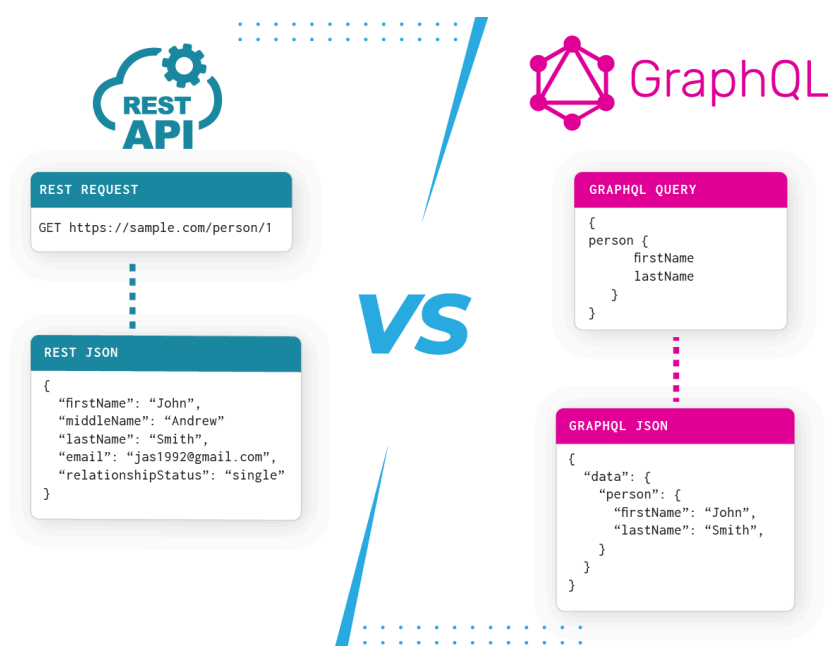
Em relação ao banco de dados, utilizaremos um banco de dados relacional, o PostgreSQL. Para gerenciar o nosso banco de dados iremos utilizar o ORM - *Object Relational Mapper* - Prisma, que já oferece uma integração com o *framework* utilizado, além dele ser bem robusto e suportar diversos bancos de dados, para prevenir futuras mudanças.

3.3.3 GraphQL

Para o desenvolvimento deste software, optou-se por não adotar a arquitetura REST (*Representational State Transfer*). Em seu lugar, será empregada a linguagem de consulta *GraphQL*, criada pelo Facebook em 2012 e disponibilizada ao público em 2015. Conforme documentado em sua especificação oficial (GRAPHQL, 2017), o *GraphQL* é reconhecido como uma alternativa moderna às arquiteturas REST, oferecendo não apenas um modelo de consulta mais flexível, mas também um serviço runtime capaz de interpretar e executar operações diretamente sobre uma API.

A adoção do *GraphQL* proporciona importantes vantagens arquiteturais, sobretudo no que se refere à eficiência na comunicação entre cliente e servidor. Diferentemente do REST, cuja estrutura pode resultar no problema de *overfetching* — situação em que o cliente recebe mais dados do que efetivamente necessita — o *GraphQL* permite a formulação de consultas declarativas altamente precisas, solicitando apenas os campos indispensáveis para cada operação. Esse modelo reduz a quantidade de dados trafegados, melhora a performance global da aplicação e promove maior controle sobre a estrutura de resposta. Assim, a utilização do *GraphQL* se mostra particularmente adequada para sistemas que exigem flexibilidade, otimização no consumo de dados e redução de sobrecarga nas interações com o *backend*.

Figura 9 - Comparativo entre consultas REST e GraphQL



Fonte: Cosmic Blog, 2021⁴

⁴ Disponível em: <<https://www.cosmicjs.com/blog/graphql-vs-rest-a-quick-guide>> Acesso em abr. 2022.

3.3.4 Armazenamento de imagens e vídeos

Um dos requisitos funcionais do sistema consiste na obrigatoriedade de que cada fiscalização possua registros audiovisuais — como fotografias e vídeos — do local em que foi realizada. Essa exigência visa assegurar a fidedignidade das informações inseridas no formulário eletrônico de fiscalização, possibilitando a verificação da correspondência entre os dados declarados e as evidências visuais obtidas em campo.

3.3.5 *Wasabi*

Para atender a esse requisito de forma eficiente e segura, optou-se pela utilização do *Wasabi Hot Cloud Storage*, um serviço de armazenamento em nuvem compatível com o protocolo S3 da *Amazon Web Services* (AWS). De acordo com Wasabi Technologies (2024), o serviço foi projetado para oferecer alta performance, segurança na integridade dos dados e baixo custo operacional no armazenamento de grandes volumes de informações.

O *Wasabi* destaca-se por manter uma arquitetura compatível com as soluções tradicionais de mercado, como o Amazon S3, porém com estrutura de custos mais previsível e acessível (WASABI TECHNOLOGIES, 2024). Assim, a escolha dessa tecnologia baseou-se principalmente em seu excelente custo-benefício, especialmente quando comparada a alternativas amplamente utilizadas, como o Microsoft Azure Blob Storage e o Google Cloud Storage (MICROSOFT, 2024; GOOGLE, 2024).

3.3.6 Arquitetura de envio de arquivos

Com o intuito de evitar que o usuário necessite aguardar o carregamento completo de cada imagem ou vídeo antes de realizar o envio de um novo arquivo, adotou-se um modelo assíncrono de upload. Nesse modelo, o aplicativo de fiscalização realiza o envio do arquivo e, em resposta, recebe um identificador de transação (ID). A partir desse identificador, o aplicativo é capaz de monitorar o

status de processamento do arquivo junto ao *backend*, sem interromper o fluxo de uso do sistema.

O *backend* é responsável por processar cada arquivo individualmente, de forma ordenada e escalável, por meio de uma fila de processamento assíncrona. Nessa arquitetura, os arquivos são inseridos em uma fila de mensagens, que atua como intermediária entre o envio e o armazenamento final do arquivo na nuvem. Esse mecanismo permite que o backend processe os arquivos de maneira concorrente e desacoplada, evitando sobrecargas no servidor e garantindo maior estabilidade do sistema.

Para a implementação dessa arquitetura, foram utilizadas as ferramentas *Bull.js* e *Redis*. O *Bull.js* é uma biblioteca Node.js voltada para o gerenciamento de filas de tarefas baseadas em Redis, amplamente utilizada por sua eficiência, robustez e suporte a operações concorrentes (BULL.JS, 2024). Já o *Redis* é um sistema de armazenamento em memória de alta performance, empregado como repositório das filas e estados de processamento, garantindo baixa latência e alta disponibilidade (REDIS, 2024). Após o processamento de cada tarefa, o arquivo é enviado ao serviço de armazenamento em nuvem Wasabi, assegurando a integridade, durabilidade e acessibilidade dos dados enviados pelos usuários.

A adoção do modelo assíncrono apresenta diversas vantagens em relação às abordagens síncronas tradicionais. Dentre elas, destacam-se a melhoria na experiência do usuário, que não precisa aguardar o término do upload de cada arquivo; o aumento da escalabilidade, permitindo o processamento simultâneo de múltiplas requisições; e a resiliência da aplicação, uma vez que falhas temporárias podem ser tratadas automaticamente pelo sistema de filas, sem perda de dados.

3.3.7 Geração de formulários e relatórios

A partir dos dados coletados pelo fiscal *in loco*, são gerados os formulários e relatórios em formato PDF, que posteriormente são armazenados no serviço de armazenamento Wasabi. Esse procedimento assegura a preservação, organização e disponibilidade dos documentos, permitindo sua utilização posterior em processos de consulta, auditoria e supervisão.

Para viabilizar a geração desses documentos de forma estruturada e padronizada, o software utiliza a ferramenta BIRT (Business Intelligence and Reporting Tools), uma solução de código aberto amplamente empregada na criação e formatação de relatórios customizados. Integrado ao ecossistema Eclipse, o BIRT possibilita o desenvolvimento de modelos robustos a partir de diferentes fontes de dados, permitindo a produção de documentos em formatos como PDF, HTML e outros padrões amplamente utilizados em ambientes corporativos.

Com uma interface gráfica intuitiva, o BIRT oferece recursos avançados para construção de layouts complexos, como tabelas, gráficos, agrupamentos, seções dinâmicas e elementos visuais que enriquecem a apresentação das informações. A ferramenta também permite a definição de parâmetros, aplicação de regras condicionais e manipulação dos dados durante a geração dos relatórios, proporcionando alta flexibilidade e aderência às necessidades específicas de cada processo.

No contexto das atividades de fiscalização e gestão pública realizadas no âmbito do PNCF, o BIRT destaca-se por sua capacidade de produzir relatórios estruturados e uniformes, garantindo que as informações coletadas em campo sejam apresentadas de maneira clara, organizada e alinhada às exigências institucionais. Sua integração com APIs, bancos de dados e sistemas de *backend* reforça sua eficácia como solução para automação de documentos oficiais, abrangendo formulários, pareceres, laudos e demais registros administrativos.

3.4 FRONTEND

O frontend corresponde à camada de apresentação da aplicação, sendo responsável pela interface gráfica acessada e manipulada diretamente pelo usuário. Seu principal objetivo é proporcionar uma interação eficiente, intuitiva e responsiva, traduzindo visualmente as funcionalidades e dados processados pelo backend.

No software desenvolvido, essa camada desempenha um papel fundamental na operacionalização do controle e gerenciamento das fiscalizações, disponibilizando recursos para registrar, acompanhar e visualizar informações relacionadas às atividades realizadas em campo. Por meio da interface gráfica, o

usuário pode acessar projetos aprovados, criar demandas de fiscalização, monitorar diferentes status e emitir relatórios de acompanhamento.

Além de contribuir para a usabilidade, o frontend impacta diretamente a experiência do usuário (UX) e a eficiência operacional do sistema, garantindo que as informações sejam apresentadas de maneira organizada, acessível e coerente com os objetivos da plataforma.

No frontend, manteve-se a utilização do TypeScript, já empregado na camada de backend, de modo a garantir maior padronização e consistência entre os diferentes módulos do sistema. Essa continuidade tecnológica favorece a integração entre as camadas, assegurando coerência na definição de tipos e contratos de dados, além de aprimorar a manutenção e a qualidade geral do código, especialmente em um ambiente baseado em React e Next.js, onde a tipagem contribui para interfaces mais estáveis e seguras.

3.4.1 Next.js e reactJS no desenvolvimento do frontend

O desenvolvimento da camada de apresentação *web* do sistema foi estruturado com base no uso combinado do *ReactJS* e do *Next.js*, tecnologias amplamente consolidadas no ecossistema *JavaScript* moderno e reconhecidas por sua eficiência, flexibilidade e elevada capacidade de composição de interfaces complexas. A adoção dessas ferramentas visa garantir uma experiência de usuário robusta, responsiva e alinhada às necessidades operacionais do sistema de fiscalização.

O *ReactJS* é uma biblioteca *JavaScript* de código aberto destinada à construção de interfaces de usuário dinâmicas e reativas. Mantido pelo *Facebook*, *Instagram* e por uma extensa comunidade de desenvolvedores, o *React* destaca-se por sua abordagem baseada em componentes, na qual a interface é dividida em estruturas modulares, reutilizáveis e independentes. Essa característica promove maior organização, facilita a manutenção do código e reduz a duplicação de lógica, contribuindo para um desenvolvimento mais sustentável e escalável. Sua ampla adoção por empresas de grande porte — como *Netflix*, *Airbnb*, *Walmart*, *Feedly* e *Imgur* (REACT, 2022) — evidencia sua maturidade tecnológica e confiabilidade no mercado.

Além da componentização, o React implementa um mecanismo de atualização eficiente denominado Virtual DOM, responsável por otimizar a renderização e garantir maior desempenho ao atualizar apenas partes da interface que sofreram alterações. Esse modelo reduz o custo computacional e aumenta a responsividade, especialmente em aplicações que realizam mudanças frequentes na UI, como sistemas de monitoramento, dashboards e plataformas interativas.

O *Next.js* surge como um framework baseado em React que expande significativamente suas capacidades, incorporando funcionalidades avançadas que visam otimizar tanto a experiência do usuário quanto o desempenho global da aplicação. Entre suas principais contribuições está o suporte nativo a diferentes estratégias de renderização, como *Single Page Application* (SPA), *Server-Side Rendering* (SSR) e *Static Site Generation* (SSG).

A abordagem SPA permite que a aplicação carregue apenas uma página inicial e atualize seu conteúdo dinamicamente, proporcionando uma navegação fluida e contínua, semelhante à experiência de aplicativos *desktop*. Em contrapartida, o SSR realiza a renderização das páginas diretamente no servidor antes de entregá-las ao navegador, resultando em tempos de carregamento mais rápidos e melhor desempenho em dispositivos com menor capacidade de processamento. Já o SSG permite gerar páginas estáticas em tempo de build, oferecendo desempenho ainda mais elevado em conteúdos que não exigem atualizações frequentes.

O *Next.js* também oferece uma série de mecanismos integrados que simplificam a estrutura do projeto, tais como roteamento automático baseado em arquivos, pré-carregamento de páginas e otimizações inteligentes de carregamento de recursos. Esses recursos tornam o desenvolvimento mais produtivo e reduzem a necessidade de configurações manuais complexas, que seriam necessárias caso se utilizasse apenas o React em sua forma tradicional.

Outro ponto relevante é a integração fluida com *TypeScript*, que contribui para maior segurança, previsibilidade e consistência na escrita do código. Essa integração é particularmente importante em sistemas corporativos e governamentais, como o proposto, onde a precisão no tratamento de dados e a estabilidade da interface são fatores fundamentais para garantir confiabilidade operacional.

A escolha conjunta de *ReactJS* e *Next.js*, portanto, oferece uma base sólida para a construção do *frontend* da aplicação, combinando modularidade, escalabilidade, desempenho e moderno padrão de desenvolvimento. Essas tecnologias possibilitam a criação de interfaces claras, responsivas e altamente interativas, capazes de atender às demandas funcionais e não funcionais do sistema de fiscalização, garantindo ao usuário final uma experiência eficiente, estável e coerente com os objetivos do projeto.

3.5 MOBILE

A parte mobile do *software* corresponde à aplicação desenvolvida com o objetivo de viabilizar as fiscalizações realizadas em campo pelos agentes responsáveis. Por meio dessa aplicação, os fiscais poderão registrar informações diretamente nos assentamentos, inserindo dados sobre as condições do local, o uso da terra e o cumprimento das metas estabelecidas pelo programa.

Essa abordagem permite maior agilidade, precisão e segurança na coleta das informações, eliminando a necessidade de formulários em papel e reduzindo o tempo entre a realização da fiscalização e o envio dos dados para análise.

Além disso, a aplicação mobile foi planejada para funcionar de forma integrada ao sistema web e ao *backend*, garantindo a sincronização automática dos dados sempre que houver conexão com a internet. Essa integração proporciona uma comunicação contínua entre os fiscais em campo e as UTEs, permitindo um acompanhamento em tempo real das atividades de fiscalização e facilitando a gestão dos projetos financiados.

3.5.1 Dart e flutter

Para o desenvolvimento da aplicação móvel, optou-se pela utilização da linguagem *Dart*, criada e mantida pela empresa *Google*. Essa escolha se justifica, principalmente, pelo uso do *framework Flutter*, também desenvolvido pela *Google*, amplamente reconhecido como uma das soluções mais eficientes e modernas para o desenvolvimento de aplicações móveis multiplataforma.

O Flutter permite que o desenvolvedor produza, a partir de um único código-fonte, aplicações nativas para os sistemas operacionais *Android* e *iOS*, que representam as plataformas móveis mais utilizadas no mundo. Essa característica elimina a necessidade de equipes distintas para cada sistema, otimizando o tempo de desenvolvimento e reduzindo os custos de manutenção e atualização do software.

Além da produtividade e da portabilidade, outro fator determinante para a escolha do Flutter é sua alta performance. Diferentemente de outros frameworks multiplataforma, como o *React Native*, o Flutter compila o código diretamente para código nativo, permitindo acesso direto ao hardware do dispositivo. Essa abordagem elimina a dependência de camadas intermediárias de interpretação, resultando em maior desempenho, menor latência e uma experiência de uso mais fluida para o usuário final.

Assim, a combinação entre a linguagem *Dart* e o *framework Flutter* proporciona uma solução robusta, eficiente e moderna, alinhada às necessidades de um sistema que exige confiabilidade, responsividade e compatibilidade entre diferentes plataformas móveis.

4 SISTEMA DE FISCALIZAÇÃO DO PROGRAMA NACIONAL DE CRÉDITO FUNDIÁRIO

Este capítulo tem como finalidade apresentar, de forma sistemática e fundamentada, os resultados alcançados por meio do desenvolvimento e da implementação do software descrito ao longo deste relatório técnico. Busca-se evidenciar os principais avanços obtidos, bem como demonstrar a efetividade das soluções propostas diante dos objetivos previamente estabelecidos.

O sistema desenvolvido reúne um conjunto de funcionalidades voltadas à gestão e execução das fiscalizações do PNCF. Os usuários autenticados podem acessar a plataforma e utilizar as funcionalidades conforme seus perfis de permissão. Administradores e gestores de UTEs podem cadastrar novos usuários, registrar e gerenciar UTEs, criar assentamentos e vinculá-los às respectivas unidades, além de incluir parcelas e lotes individualmente ou em lote (*batch*).

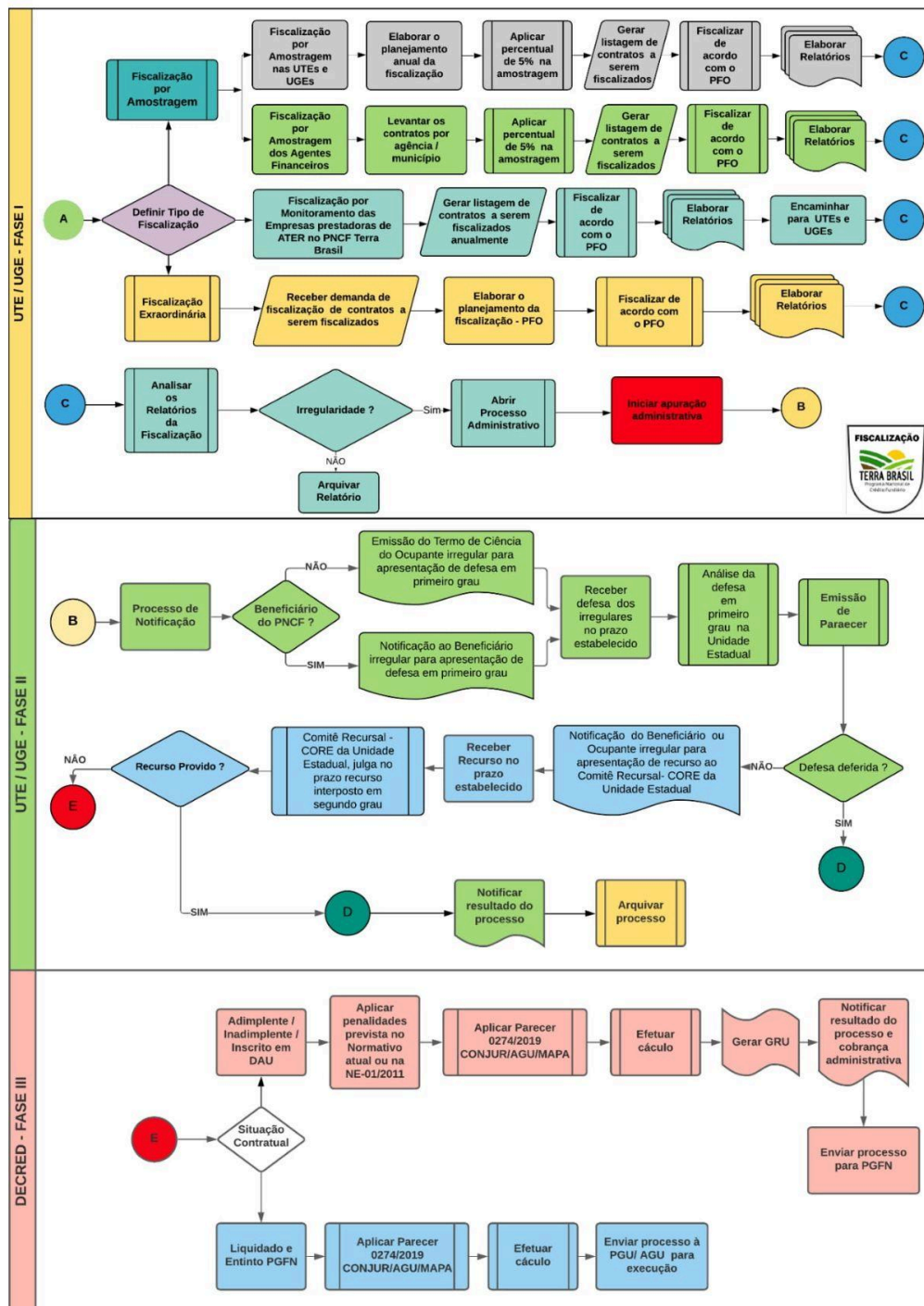
O *software* permite a designação de fiscais às fiscalizações registradas, garantindo que esses agentes possam realizar as vistorias tanto em modo online quanto offline, assegurando continuidade das operações mesmo em áreas com baixa conectividade. Além disso, a plataforma oferece recursos de listagem e gerenciamento de assentamentos, parcelas e lotes, facilitando o acompanhamento organizado das informações. Por fim, todos os usuários podem realizar a alteração de suas senhas, reforçando a segurança e autonomia no acesso ao sistema.

4.1 FLUXOS DE FISCALIZAÇÃO

A Figura a seguir apresenta o fluxo ideal de funcionamento previsto para o software, ilustrando de forma clara as etapas, interações e decisões que compõem o processo operacional do sistema.

Figura 10 - Fluxos de fiscalização

Fluxo dos Processos de Fiscalização da Regularidade Ocupacional e Exploração Efetiva do PNCF Terra Brasil



Fonte: Manual de fiscalização do PNCF

4.2 REQUISITOS

A definição dos requisitos é fundamental para orientar o desenvolvimento da solução proposta, assegurando que todas as funcionalidades necessárias para o apoio às fiscalizações do PNCF sejam contempladas de maneira clara, objetiva e verificável. Nesta seção, são apresentados os requisitos funcionais e não funcionais identificados ao longo da análise do problema, considerando as necessidades dos usuários, os fluxos de trabalho das UTEs e as particularidades das atividades executadas pelos fiscais em campo.

4.2.1 Requisitos Funcionais

Os requisitos funcionais descrevem o conjunto de comportamentos e funcionalidades que o sistema deve oferecer para atender às operações previstas no processo de fiscalização. Entre os principais requisitos, destacam-se:

- **RF01 – Gestão de UTEs:** o sistema deve permitir o cadastro, atualização e gerenciamento de Unidades Técnicas Estaduais.
- **RF02 – Designação de Fiscais:** usuários com perfil fiscal devem ser vinculados às fiscalizações registradas, conforme demanda operacional.
- **RF03 – Execução de Fiscalizações Online e Offline:** o sistema deve permitir que fiscais realizem vistorias mesmo sem conexão, garantindo sincronização posterior dos dados.
- **RF04 – Listagem e Gerenciamento de Assentamentos, Parcelas e Lotes:** o sistema deve permitir a consulta e administração dessas entidades de forma organizada.

4.2.2 Requisitos Não Funcionais

Os requisitos não funcionais estabelecem características de qualidade essenciais para garantir eficiência, segurança e confiabilidade na utilização do sistema.

- **RNF01 – Usabilidade:** a interface deve ser intuitiva e de fácil navegação para usuários com diferentes níveis de familiaridade tecnológica.
- **RNF02 – Confiabilidade:** as informações devem ser armazenadas de forma íntegra e protegida contra perdas ou inconsistências.
- **RNF03 – Tolerância à Falha e Operação Offline:** o sistema deve ser capaz de operar em locais sem conectividade e garantir sincronização segura quando a conexão for restabelecida.

4.2.3 Casos de uso

Os requisitos apresentados nas seções anteriores serviram de base para a definição dos casos de uso descritos a seguir. Nesta etapa, são apresentados apenas os principais casos de uso, selecionados por representarem as interações mais relevantes entre os usuários e o sistema. Os demais casos de uso, igualmente importantes para a compreensão completa do funcionamento da aplicação, estão disponíveis na seção de anexos deste documento.

Tabela 01 - Cadastrar Unidade Técnicas Estaduais (UTE)

C-01	Cadastrar Unidade Técnicas Estaduais (UTE)
Ator Principal	Administrador.
Descrição	O usuário administrador deverá preencher o formulário para o cadastro de UTEs.

Propósito	Gerenciar os assentamentos daquele estado.
Pré-Condição	O usuário deve estar cadastrado.
Pós-Condição	Retornar a UTE cadastrada.
Fluxo Principal	1. Preencher o formulário; 2. Confirmar o formulário de cadastro de UTE; 4. Acessar a aba de UTEs; 5. Preencher o formulário; 6. Confirmar o formulário de cadastro de UTE;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de UTEs; 5. Preencher o formulário; 6. Confirmar o formulário de cadastro de UTE; 7. Já existe um UTE naquele estado;
Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Preencher o formulário; 5. Não foi possível estabelecer conexão com o sistema; 6. Mostrar o erro ao usuário;

Tabela 02 - Incluir Agentes Fiscais em uma fiscalização

C-02	Incluir Agentes Fiscais em uma fiscalização
Ator Principal	Administrador, Gestor de UTE.
Descrição	O usuário administrador ou gestor de UTE deverá preencher o formulário para a inclusão de um fiscal em um fiscalização.
Propósito	Realizar fiscalização naquele assentamento.
Pré-Condição	O assentamento deverá estar cadastrado.
Pós-Condição	
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar abe assentamentos 5. Selecionar um assentamento. 6. Preencher o formulário; 7. Confirmar o formulário de inclusão de fiscal;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar abe assentamentos 5. Selecionar um assentamento. 6. Preencher o formulário; 7. Fiscal já incluso naquela fiscalização;

Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. Utilizar o sistema; 4. Acessar abe assentamentos 5. Selecionar um assentamento. 6. Preencher o formulário; 2. Não foi possível estabelecer conexão com o sistema; 3. Mostrar o erro ao usuário;

Fonte: Autoria Própria

Tabela 03 - Criar fiscalização

C-03	Criar fiscalização
Ator Principal	Administrador, Gestor de UTE.
Descrição	O administrador ou gestor de UTE, poderão criar fiscalizações extraordinárias ou por amostragem.
Propósito	Verificar se os assentamentos estão cumprindo os requisitos do Programa Nacional de Crédito Fundiário ou se estão com alguma irregularidade.
Pré-Condição	Ter acesso ao sistema.
Pós-Condição	
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. Utilizar o sistema;

	4. Acessar a aba de assentamento; 5. Selecionar um assentamento; 6. preencher o formulário para iniciar uma fiscalização;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. Utilizar o sistema; 4. Acessar a aba de assentamento; 5. Selecionar um assentamento; 6. Já existe uma fiscalização em andamento naquele assentamento;
Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. Utilizar o sistema; 4. Acessar a aba de assentamento; 5. Selecionar um assentamento; 6. Não foi possível estabelecer conexão com o sistema; 7. Mostrar o erro ao usuário;

Fonte: Autoria Própria

4.3 APLICATIVO MÓVEL DE FISCALIZAÇÃO

As Figuras apresentadas nas próximas seções têm como finalidade demonstrar, de forma abrangente e detalhada, a interface e o funcionamento da aplicação móvel desenvolvida para o PNCF. Essas representações visuais permitem uma compreensão ampliada não apenas dos elementos gráficos que compõem cada tela, mas também das interações possíveis, dos fluxos de navegação e da lógica operacional que sustenta o uso da aplicação em campo.

Por meio das imagens, é possível observar a organização dos componentes da interface, a disposição dos menus, botões e formulários, bem como a aplicação

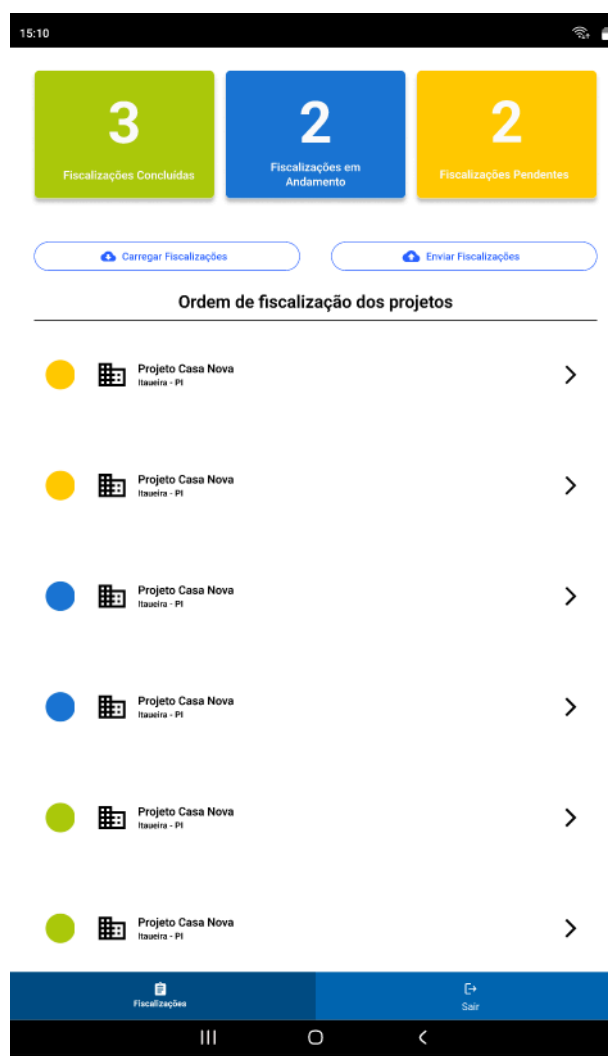
de princípios de usabilidade adotados durante o processo de desenvolvimento, tais como clareza, consistência visual, acessibilidade e otimização da experiência do usuário. Além disso, as figuras evidenciam como o design da aplicação foi pensado para atender às particularidades do público-alvo — em especial, fiscais que frequentemente realizam suas atividades em regiões rurais, muitas vezes com alta luminosidade e necessidade de agilidade na coleta de dados.

A apresentação das telas também permite compreender de que maneira o sistema integra os mecanismos de operação offline, exibindo, por exemplo, indicadores de sincronização, estruturas de armazenamento local temporário e fluxos que asseguram que os dados coletados em campo sejam devidamente preservados até que haja conexão disponível para transmissão. Esses aspectos são fundamentais para garantir a continuidade das atividades de fiscalização, reforçando a confiabilidade e a robustez da aplicação em cenários adversos.

Adicionalmente, as figuras auxiliam na visualização das etapas que compõem a realização de uma fiscalização, desde o acesso inicial do usuário até o preenchimento das informações, anexação de evidências e finalização do processo. Tal abordagem contribui para a validação prática dos requisitos funcionais estabelecidos, uma vez que permite demonstrar como cada funcionalidade foi implementada e como as interações foram estruturadas para proporcionar uma experiência coerente e eficiente.

Dessa forma, o conjunto de imagens apresentado neste capítulo desempenha papel essencial na contextualização e na comprovação do funcionamento da solução desenvolvida, oferecendo uma visão concreta da aplicação e complementando a descrição técnica detalhada ao longo deste documento.

Figura 11 - Tela inicial da aplicação móvel



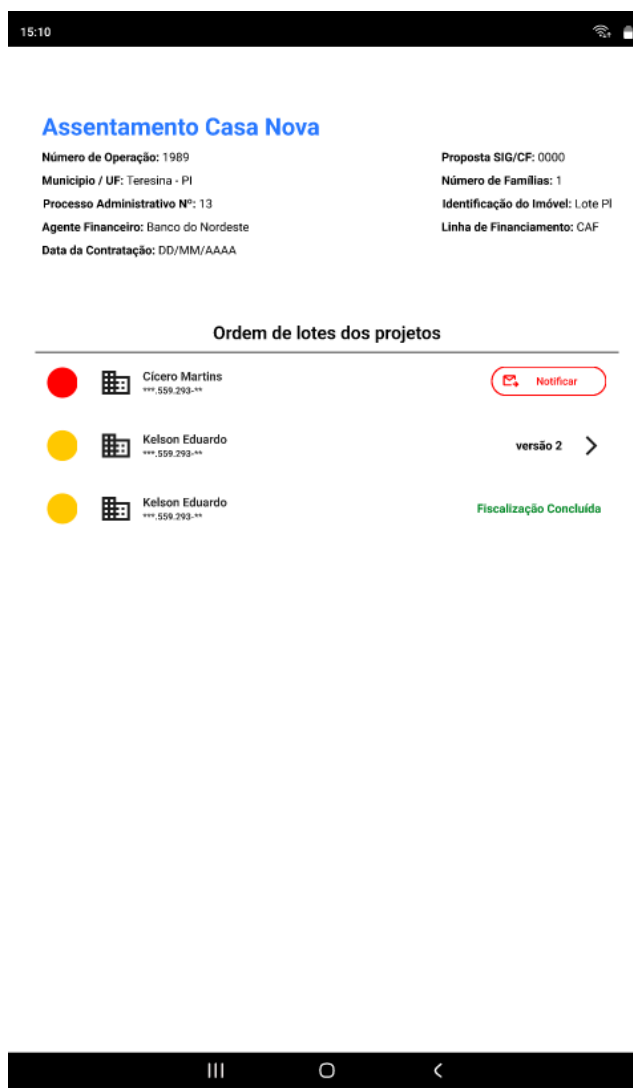
Fonte: Autoria Própria

A Figura acima corresponde ao painel principal da aplicação móvel do PNCF, oferecendo ao fiscal uma visão consolidada das fiscalizações sob sua responsabilidade. Na parte superior, três cartões indicam a quantidade de fiscalizações concluídas, em andamento e pendentes, utilizando cores distintas para facilitar a identificação. A interface também disponibiliza botões para carregar e enviar fiscalizações, permitindo a sincronização dos dados conforme a disponibilidade de conexão.

Abaixo, é exibida a lista organizada de projetos a serem fiscalizados, apresentando o nome do assentamento, sua localização e um indicador visual de status. Cada item permite acesso direto à fiscalização correspondente. Na parte inferior, a barra de navegação oferece acesso rápido ao módulo de fiscalizações e à

opção de logout. A disposição dos elementos demonstra foco em usabilidade, clareza das informações e suporte às atividades em campo.

Figura 12 - Tela principal de uma assentamento



Fonte: Autoria Própria

A Figura apresentada exhibe os detalhes de um assentamento selecionado no módulo de fiscalizações do PNCf. Na parte superior, são mostradas as principais informações administrativas do projeto, incluindo número de operação, município, processo administrativo, agente financeiro e dados do imóvel, permitindo ao fiscal consultar rapidamente o contexto da vistoria. Abaixo, a seção “Ordem de lotes dos projetos” lista os lotes vinculados ao assentamento, acompanhados por indicadores visuais de status, nome do beneficiário e identificação resumida.

Cada item da lista apresenta ações específicas conforme o estado da fiscalização: alguns lotes possuem a opção de notificação, enquanto outros indicam a disponibilidade de uma nova versão da vistoria ou mostram o status de “Fiscalização Concluída”. Um ícone de navegação permite acessar diretamente o registro detalhado do lote selecionado. A organização dessa tela facilita o acompanhamento das atividades associadas a cada lote, oferecendo ao fiscal uma visão clara do progresso e das pendências relacionadas ao assentamento.

Figura 13 - Aba de operação de financiamento

15:10

Operação de Financiamento Dados dos Beneficiários Tipificação e Ocupantes Irregulares Infraestruturas do Lote Exploração Efetiva C

Operação de Financiamento

Número do lote	Área do Lote hectare
Projeto Coletivo	CNPJ
ASSOCIAÇÃO DE DESENVOLVIMENTO COMUNITÁRIO DO AS	59 456 277/0001-76
Identificação do Imóvel	Número de Famílias
Não Informado	0
Município	Estado
MORRO DO CHAPEU DO PIALUI	PI
Número Único da Operação	Proposta SIG/CF
-	-
Linha de Financiamento	Data de Contratação
PROGRAMA DE COMBATE A POBREZA	13 de Setembro de 2022
Processo Administrativo Nº	Agente Financeiro
PROGRAMA DE COMBATE A POBREZA	Banco do Nordeste do Brasil S.A

Cancelar Salvar

III O <

Fonte: Autoria Própria

Figura 14 - Aba de dados do beneficiário

15:10

Operação de Financiamento

Dados dos Beneficiários

Tipificação e Ocupantes Irregulares

Infraestruturas do Lote

Exploração Efetiva

Dados do Beneficiário

Nome do Titular 01

CPF

RG

Nome do Titular 02

CPF

RG

Cadastro no CAD Único

Não

Sim

Cadastro Bolsa Família

Não

Sim

Possui Filhos ou Dependentes

Não

Sim

Número de Filhos

-

0

+

Outros Dependentes

-

0

+

Contatos

Telefone

e-mail

Mesmo número de WhatsApp?

Não

Sim

WhatsApp

Endereço para correspondência

Longradouro

Bairro

Município

Unidade Federativa - UF

Poço

Poço

Observações

Observações sobre os beneficiários:

digite as observações aqui

Cancelar

Salvar

III

O

<

Fonte: Autoria Própria

Figura 15 - Aba de tipificação e ocupação irregular

15:10

Operação de Financiamento Dados dos Beneficiários **Tipificação e Ocupantes Irregulares** Infraestruturas do Lote Exploração Efetiva

Tipificação e Ocupantes Irregulares

Dados Gerais do Lote

Parcela Regular?

☒ Não ☐ Sim

Situação de Ocupação da Parcela

Escolha uma opção

Observações

Observações sobre a tipificação da parcela:

digite as observações aqui

Cancelar Salvar

Fonte: Autoria Própria

Figura 16 - Aba de infraestrutura do lote

Fonte: Autoria Própria

As Figuras 12, 13, 14 e 15 apresentam de forma organizada o funcionamento do módulo de fiscalização da aplicação móvel, este tópico destaca apenas as principais abas que compõem o processo de registro das informações coletadas em campo. A seleção dessas abas busca fornecer uma visão geral da estrutura do sistema, ilustrando como o fluxo da fiscalização é distribuído entre etapas distintas e sequenciais. Cada aba reúne um conjunto específico de informações, conduzindo o fiscal por diferentes dimensões da vistoria — desde os dados administrativos da

operação até informações relacionadas aos beneficiários e às características do lote — garantindo, assim, que os requisitos essenciais do procedimento sejam devidamente contemplados durante o preenchimento.

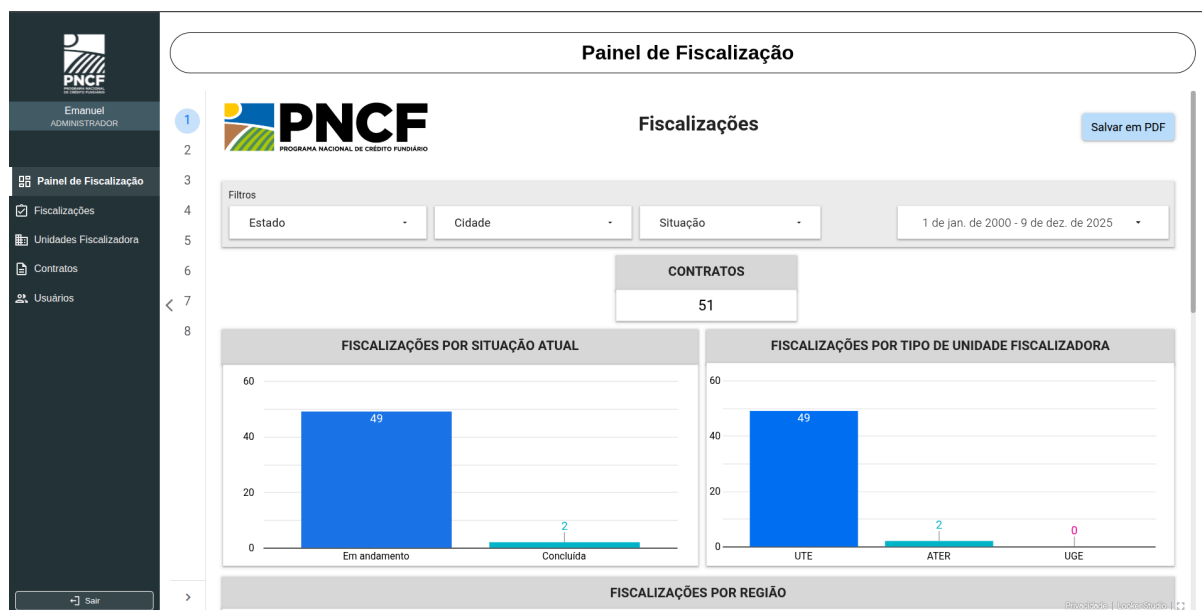
A organização modular adotada pela aplicação facilita a navegação e reduz a probabilidade de omissões, uma vez que o usuário é guiado por categorias bem definidas, projetadas para refletir a lógica operacional exigida pelo PNCF. Ainda que este capítulo apresenta apenas as abas mais representativas, a aplicação dispõe de outras seções igualmente importantes para a completude da fiscalização. Entretanto, para manter a objetividade e evitar excesso de detalhamento no corpo principal do texto, as demais abas foram incluídas no campo de anexos deste documento, onde podem ser consultadas integralmente conforme a necessidade de aprofundamento.

Dessa forma, esta apresentação busca equilibrar clareza e concisão, evidenciando o funcionamento estrutural do módulo de fiscalização sem comprometer a fluidez do capítulo, ao mesmo tempo em que disponibiliza, nos anexos, material complementar que permite uma compreensão mais abrangente da solução desenvolvida.

4.4 INTERFACE DA PLATAFORMA *WEB*

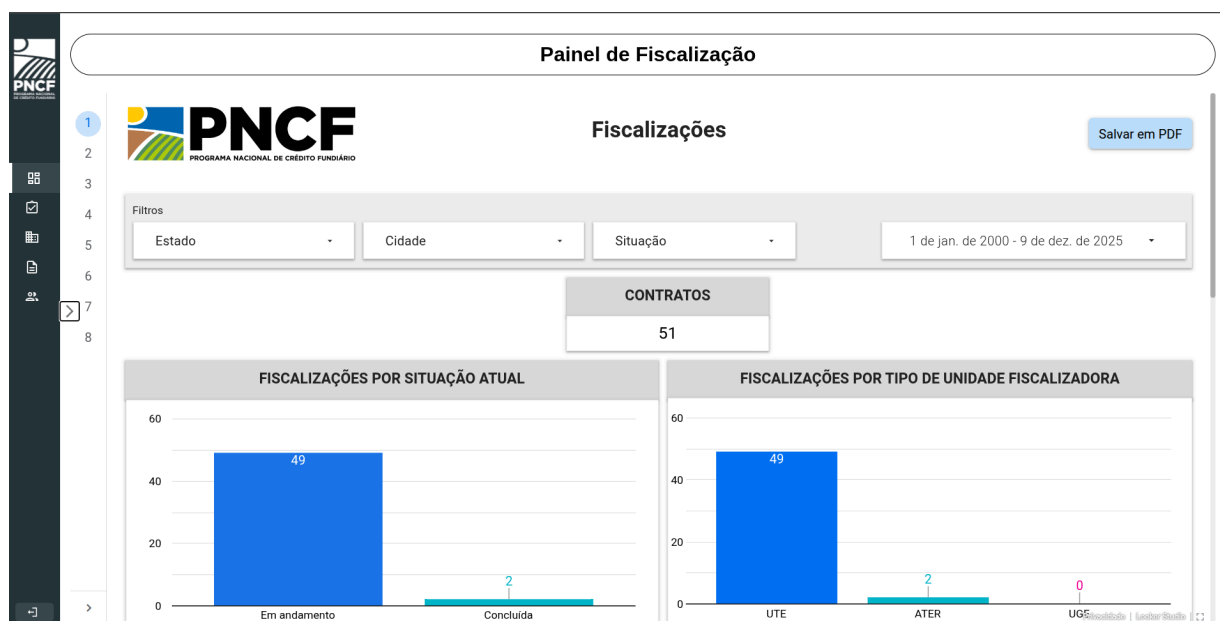
As figuras apresentadas nas próximas seções demonstram a interface e o funcionamento do painel web desenvolvido para o gerenciamento das operações do PNCF. Por meio dessas imagens, é possível visualizar a organização dos menus, tabelas e formulários que permitem administrar fiscalizações, assentamentos, UTEs e usuários, além de realizar a atribuição de vistorias aos fiscais. As telas evidenciam também recursos essenciais, como a visualização dos formulários enviados pelo aplicativo móvel e a geração de relatórios, permitindo acompanhar de forma estruturada o andamento das atividades. Dessa maneira, as representações visuais apresentadas oferecem uma visão geral da lógica administrativa do sistema e complementam a descrição técnica das funcionalidades implementadas.

Figura 17 - Painel principal da aplicação web com a barra lateral aberta



Fonte: Autoria Própria

Figura 18 - Painel principal da aplicação web com a barra lateral aberta



Fonte: Autoria Própria

As figuras acima apresentam o painel principal da aplicação, no qual administradores, gestores de UTEs e fiscais podem visualizar as informações relacionadas às suas atividades. Esse dashboard reúne indicadores, atalhos e elementos de navegação que oferecem uma visão geral das operações em

andamento, facilitando o acompanhamento e a execução das tarefas de cada perfil de usuário.

Figura 19 - Listagem de Usuários

UTE	Nome	E-mail	Perfil	Ativo	Ações
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	
Alagoas Iteral	Cícero Henrique de Sousa Silva	cicero282@hotmail.com	ADMIN	☐	

1-5 de 5 < 1 >

Fonte: Autoria Própria

Tanto os administradores do sistema quanto os gestores de UTEs possuem permissões para gerenciar usuários. Os administradores realizam esse gerenciamento de forma abrangente, abrangendo todos os usuários cadastrados na plataforma, enquanto os gestores de UTEs são responsáveis especificamente pela administração dos fiscais vinculados à sua unidade. A figura acima ilustra como essa interface é apresentada para ambos os perfis, evidenciando as funcionalidades disponíveis de acordo com cada papel.

Figura 20 - Listagem de UTEs

Sigla	Nome	UF	Ações
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓
SSP	Alagoas lorem ipsum lorem ipsum	AL	✓

1-5 de 5 < 1 >

Fonte: Autoria Própria

A figura ilustra a tela de Listagem de UTEs - Unidades Técnicas Estaduais, que faz parte do painel administrativo da aplicação web. Esta interface é crucial para o gerenciamento das entidades responsáveis pela execução do PNCF no âmbito estadual.

Figura 21 - Listagem de Assentamentos

Nome	Município	UF	Parcelas	Detalhes
Assentamento Casa Nova	Teresina	PI	05	
Assentamento Casa Nova	Teresina	PI	05	
Assentamento Casa Nova	Teresina	PI	05	
Assentamento Casa Nova	Teresina	PI	05	
Assentamento Casa Nova	Teresina	PI	05	
Assentamento Casa Nova	Teresina	PI	05	
Assentamento Casa Nova	Teresina	PI	05	
Assentamento Casa Nova	Teresina	PI	05	

1-5 de 5 < 1 >

Fonte: Autoria Própria

A Figura exibe a interface de gerenciamento de assentamentos no painel web do sistema. Nessa tela, gestores e administradores podem visualizar a lista de assentamentos cadastrados, acompanhados de informações essenciais como nome, município, unidade federativa (UF) e quantidade de parcelas vinculadas. No topo da interface, há um campo de busca que permite localizar assentamentos de forma rápida, além dos botões “Novo assentamento”, utilizado para cadastrar novos registros, e “Filtrar”, que possibilita refinar a listagem conforme critérios específicos. No lado esquerdo, observa-se o menu lateral de navegação, que dá acesso às demais funcionalidades do sistema. Na coluna “Detalhes”, cada assentamento possui um ícone que direciona para informações mais completas sobre o projeto selecionado. A tela demonstra uma organização clara e funcional, facilitando o gerenciamento eficiente dos assentamentos dentro da plataforma.

Figura 22 - Aba de parcelas de um assentamento



TERRA BRASIL
Programa Nacional de
Crédito Fundiário



Admin

Início

Usuários

UTEs

Assentamentos

Fiscalizações

Notificações

Configurações

Sair

Assentamento Casa Nova

[+ Adicionar Situação Problema](#)

[+ Gerar Fiscalização](#)

[✔ Editar Assentamento](#)

[+ Adicionar Parcelas](#)

[✕ Remover Assentamento](#)

Número Único da Operação:

Proposta SIG/CF:

Município / UF:

Número de Famílias:

Processo Administrativo N°:

CNPJ:

Linha de Financiamento:

Agente Financeiro:

Data de Contratação:

Identificação do Imóvel:

Projeto Coletivo:

Parcelas

Problemas

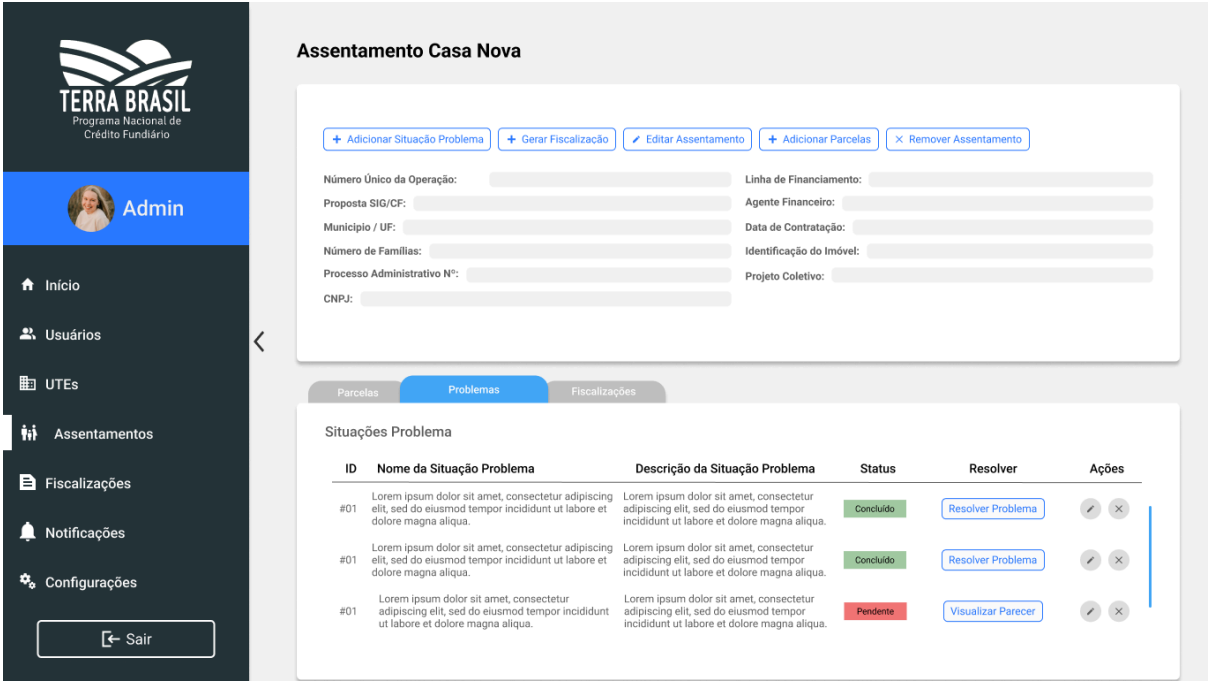
Fiscalizações

Parcelas do Assentamento

Nome	Responsável Legal	Ações
#01	João Gabriel Oliveira	
#02	Cícero Henrique Sousa	
#03	Lucas Gabriel Moraes	
#04	Mariana Gabriela Martins	
#05	Maria Vitória Soares	
#07	Beatriz Luíza Silva	

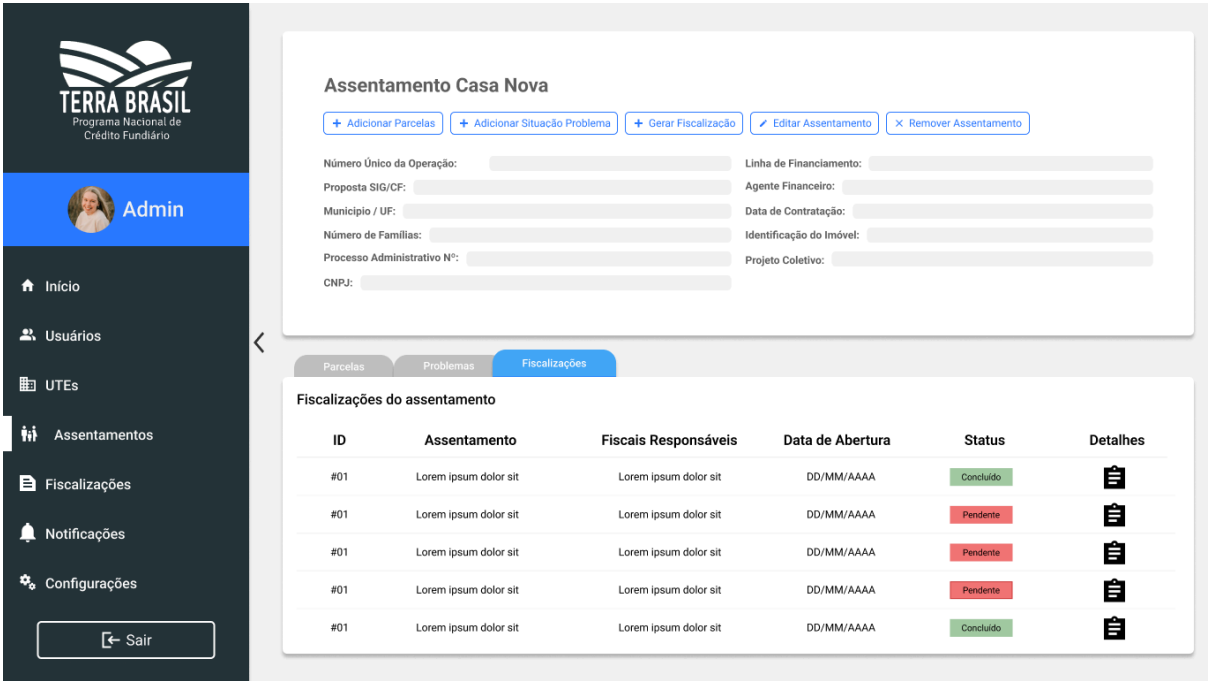
Fonte: Autoria Própria

Figura 23 - Aba de problemas de um assentamento



Fonte: Autoria Própria

Figura 24 - Aba de fiscalizações de um assentamento



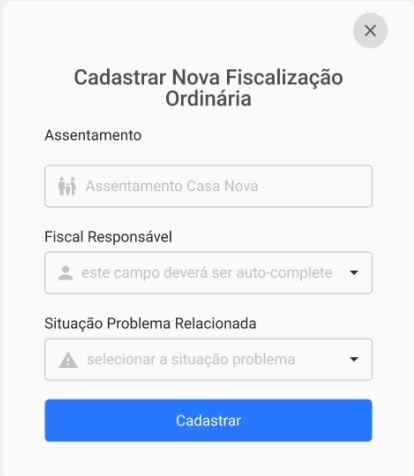
Fonte: Autoria Própria

As telas apresentadas correspondem à visualização detalhada de um assentamento no painel web do PNCF, reunindo em um único ambiente todas as informações necessárias para seu gerenciamento. Na parte superior, são exibidos os dados gerais do assentamento e as ações administrativas disponíveis, como adicionar situações problema, gerar fiscalizações, editar informações e gerenciar parcelas.

A aba Parcelas apresenta a listagem completa das parcelas vinculadas ao assentamento, acompanhada do responsável legal por cada uma delas e das ações rápidas de edição. Já a aba Problemas reúne todas as situações problema registradas, exibindo detalhes como nome, descrição, status e opções para resolução ou visualização de pareceres, permitindo o acompanhamento e tratamento das pendências do assentamento.

Por fim, a aba Fiscalizações exibe todas as fiscalizações associadas ao assentamento, apresentando informações como identificador, fiscais responsáveis, data de abertura e status da vistoria, além de acesso aos detalhes de cada fiscalização. Em conjunto, essas abas oferecem uma visão consolidada das operações relativas ao assentamento e possibilitam sua administração de forma estruturada e eficiente.

Figura 25 - Tela iniciar uma fiscalização

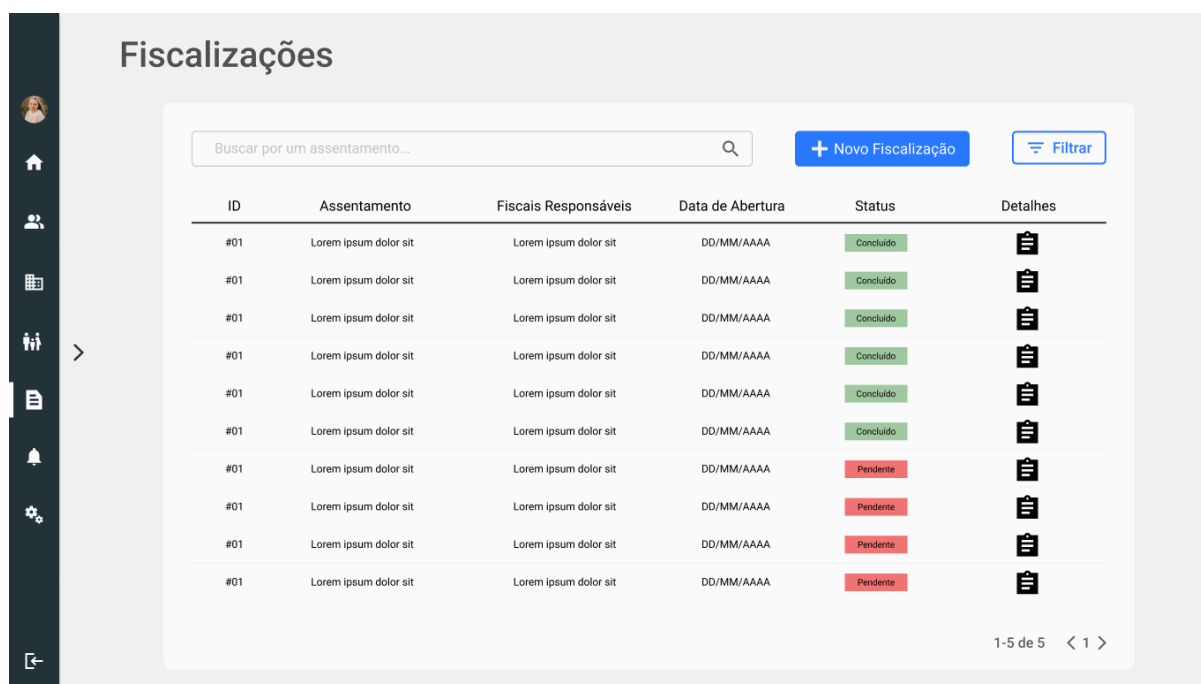


The image shows a modal window titled "Cadastrar Nova Fiscalização Ordinária" (Register New Ordinary Fiscalization). It contains three input fields: "Assentamento" (Settlement) with a dropdown menu showing "Assentamento Casa Nova"; "Fiscal Responsável" (Fiscal Responsible) with a dropdown menu showing "este campo deverá ser auto-complete"; and "Situação Problema Relacionada" (Related Problem Situation) with a dropdown menu showing "selecionar a situação problema". A blue "Cadastrar" (Register) button is at the bottom.

Fonte: Autoria Própria

A tela exibida corresponde ao modal utilizado para cadastrar uma nova fiscalização ordinária no painel *web*. Nela, o usuário seleciona as informações essenciais para criar uma fiscalização, incluindo o assentamento, os fiscais responsáveis e, quando aplicável, a situação problema relacionada. O modal apresenta ainda um botão de ação para concluir o cadastro. Trata-se de uma interface simples e objetiva, voltada para agilizar o processo de criação de novas fiscalizações dentro do sistema.

Figura 26 - Tela iniciar uma fiscalização



Fonte: Autoria Própria

A tela de listagem de fiscalizações segue o mesmo padrão visual e estrutural adotado nas demais listagens do painel web como demonstrado acima, garantindo uniformidade e facilidade de uso para o usuário. Nessa interface, são apresentadas todas as fiscalizações registradas no sistema, organizadas em formato de tabela, contendo informações essenciais como identificação da fiscalização, assentamento vinculado, fiscais responsáveis, data de abertura e status atual. Além disso, cada linha apresenta ações rápidas, permitindo acessar os detalhes completos da fiscalização ou realizar operações adicionais conforme a permissão do usuário. Essa padronização facilita a navegação, otimiza o gerenciamento das vistorias e mantém a consistência da experiência oferecida pelo sistema.

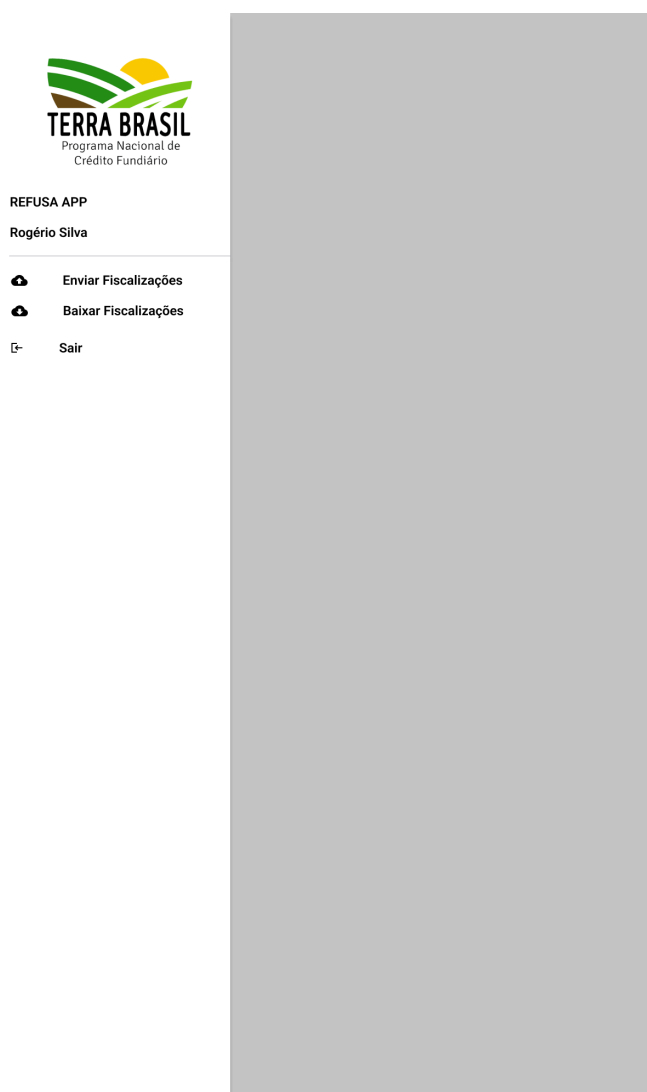
4.5 OPERABILIDADE OFFLINE

Uma das principais vantagens da aplicação móvel desenvolvida para o PNCF é sua capacidade de operar plenamente em ambientes desprovidos de acesso à internet. Essa funcionalidade constitui um dos requisitos fundamentais do sistema, permitindo que os fiscais executem suas atividades em campo de maneira contínua,

mesmo em localidades remotas com baixa ou nenhuma conectividade. A operabilidade offline garante a coleta íntegra das informações necessárias ao processo de fiscalização, assegurando que nenhum dado seja perdido e que todas as ações realizadas possam ser posteriormente sincronizadas com a infraestrutura central do sistema.

Além disso, destaca-se que as fiscalizações registradas permanecem armazenadas localmente no dispositivo utilizado pelo fiscal até que seja possível realizar o envio para o backend. Assim, o envio dos dados torna-se obrigatório tão logo o fiscal esteja em um ambiente com conexão estável, garantindo a atualização das informações no sistema e a continuidade dos fluxos administrativos do programa.

Figura 27 - Barra lateral da aplicação móvel



Fonte: Autoria Própria

A Figura acima demonstra que a barra lateral da aplicação também desempenha um papel relevante no processo de sincronização. Esse componente da interface disponibiliza ao fiscal a opção de transmitir as fiscalizações armazenadas localmente, permitindo que a sincronização seja realizada de forma prática e centralizada. Dessa maneira, o usuário pode garantir que todas as informações coletadas em campo sejam devidamente encaminhadas ao sistema assim que houver conectividade adequada.

Figura 28 - Componentes de sincronização de fiscalizações



Fonte: Autoria Própria

A Figura 00 apresenta outra seção da interface na qual o usuário pode realizar a sincronização das fiscalizações. Nesse espaço, são exibidos dois ícones principais: o primeiro, representado por uma nuvem com uma seta apontando para cima, corresponde à funcionalidade de envio dos dados armazenados localmente para o servidor, operação que ocorre assim que houver conexão disponível. O segundo ícone, uma nuvem com seta apontando para baixo, permite que o fiscal faça o download das fiscalizações antes de se deslocar para o campo, garantindo que todas as informações necessárias estejam acessíveis mesmo em ambientes sem conectividade.

4.6 BACKEND

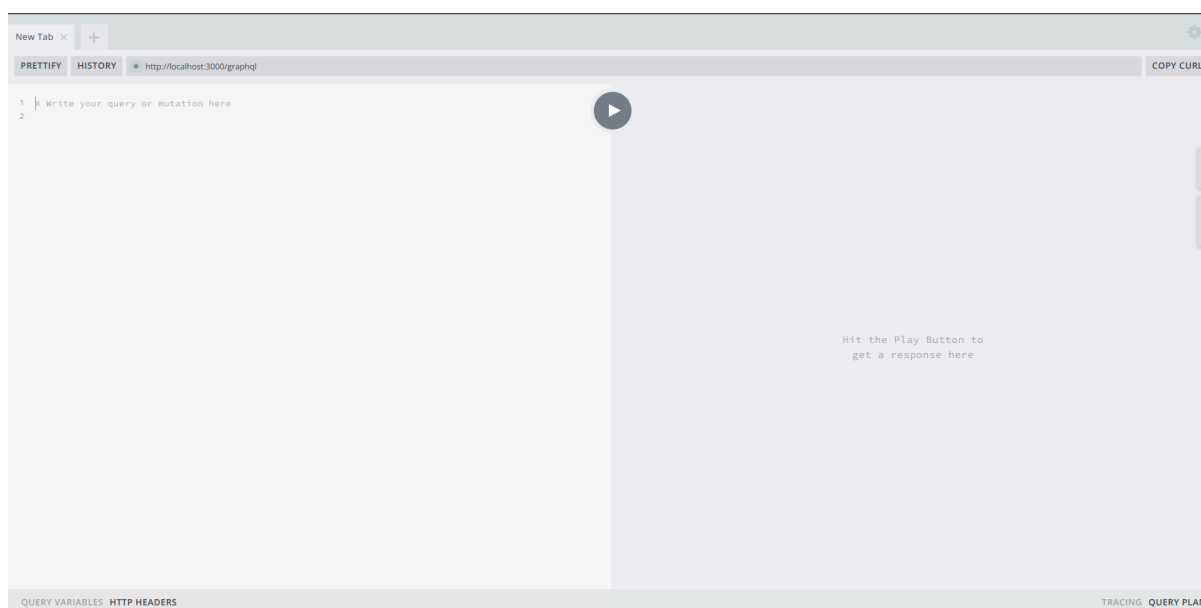
As informações apresentadas nessa seção demonstram a estrutura e o funcionamento do backend desenvolvido para sustentar as operações do sistema de fiscalização do PNCF. Por meio dessas representações, é possível visualizar a organização dos serviços disponibilizados pela API, implementada com GraphQL, bem como entender como os dados trafegam entre o painel web, o aplicativo móvel e o servidor. As telas incluem a interface do GraphQL Playground, ferramenta utilizada para inspecionar, testar e validar consultas e mutações diretamente no ambiente da API.

Além disso, são apresentados trechos do *schema GraphQL*, que definem de maneira formal os tipos, consultas, mutações e relacionamentos entre entidades como fiscalizações, assentamentos, UTEs, usuários e lotes. Esses elementos evidenciam a arquitetura do *backend*, mostrando como as informações são estruturadas, quais operações estão disponíveis aos clientes do sistema e de que forma as regras de negócio foram incorporadas de maneira padronizada e verificável.

As Figuras também exemplificam consultas e mutações utilizadas no dia a dia da aplicação, demonstrando operações como criação e atualização de usuários, cadastro de assentamentos, atribuição de fiscalizações a fiscais e sincronização de dados enviados a partir do aplicativo móvel. Essa demonstração contribui para uma compreensão prática do funcionamento interno da API, permitindo visualizar a interação entre os módulos do sistema e validar os requisitos funcionais e não funcionais definidos para o *backend*.

Dessa forma, as representações aqui apresentadas oferecem uma visão abrangente da infraestrutura lógica que suporta o sistema, evidenciando tanto sua organização quanto sua capacidade de lidar com processos complexos de gestão e fiscalização. As telas complementam a descrição técnica detalhada ao longo deste capítulo, permitindo observar de forma concreta como o backend foi construído, documentado e testado, garantindo a integridade e a confiabilidade das operações realizadas pelo PNCF.

Figura 29 - Tela inicial do graphql playground



Fonte: Autoria Própria

A tela inicial do *GraphQL Playground* apresenta um ambiente interativo destinado à exploração, teste e validação das operações disponibilizadas pela API *GraphQL* do sistema, como demonstrado na figura acima. Ao acessar o *playground*, o usuário visualiza,

à esquerda, o editor de consultas, um espaço dedicado à escrita de *queries*, *mutations* e *subscriptions*. Esse editor oferece recursos como destaque de sintaxe, autocompletar e validação automática, facilitando a navegação pelo schema e reduzindo erros durante a construção das operações.

No painel à direita, são exibidos os resultados retornados pela API em formato JSON, permitindo observar em tempo real a estrutura das respostas e validar o comportamento das funcionalidades implementadas no backend. Acima desse painel, encontram-se controles adicionais, como a opção de formatar as respostas, limpar o histórico ou reiniciar a sessão.

À direita da interface, o menu Docs disponibiliza a documentação gerada automaticamente a partir do *schema GraphQL*. Nele, é possível consultar todos os tipos, campos, consultas e mutações expostas pela API, bem como seus argumentos e estruturas de retorno. O menu Schema, por sua vez, apresenta uma visão consolidada de todo o modelo de dados, permitindo que desenvolvedores compreendam rapidamente a organização e os relacionamentos entre as entidades.

De forma geral, a tela inicial do *GraphQL Playground* fornece um ambiente completo para inspeção e validação da API, oferecendo tanto ferramentas de escrita e execução de operações quanto documentação integrada, possibilitando testar de maneira prática o funcionamento interno do backend.

Figura 30 - Código graphql que define o tipo usuário

```
type User {  
  id: String  
  createdAt: DateTime  
  updatedAt: DateTime  
  deletedAt: DateTime  
  name: String!  
  role: Role!  
  email: String!  
  phone: String  
  
  """campo matrícula"""  
  registration: String  
  active: Boolean!  
  inspectionUnit: InspectionUnit  
  cpf: String  
}
```

Fonte: Autoria Própria

Figura 31 - Código graphQL que define o tipo assentamento

```

type Settlement {
  id: String
  createdAt: DateTime
  updatedAt: DateTime
  deletedAt: DateTime
  name: String!
  origin: String
  inspectionUnit: InspectionUnit
  allotments: [Allotment!]!
  situationProblems: [SettlementSituationProblem!]!
  settlementInspection: [SettlementInspection!]!
  operationNumber: String
  area: String
  proposal: String
  contractDate: DateTime
  administrativeProcess: String
  numberOfFamilies: Float
  propertyName: String
  lot: String
  contractorCpfCnpj: String
  contractorName: String
  isCollective: Boolean
  state: State!
  city: City!
  financingLine: FinancialLine
  financialAgent: FinancialAgent
  allotmentsCount: Float!
}

```

Fonte: Autoria Própria

As duas Figuras acima ilustram como um tipo é definido na biblioteca GraphQL. Cada propriedade do tipo é declarada seguida de dois pontos, indicando em sequência o tipo de dado correspondente. Essa estrutura permite especificar, de forma explícita e padronizada, o formato das informações que compõem cada entidade no *schema*.

Figura 32 - Código graphQL para busca de assentamentos

```

query settlement($input: FindSettlementsInput!) {
  Execute Query
  settlements(input: $input) {
    totalCount
    data {
      id
      name

      numberOfFamilies
      state {
        id
        name
        uf
      }
      city {
        id
        name
        uf
      }
    }
  }
}

```

Fonte: Autoria Própria

Figura 33 - Código graphQL para criação de usuários

```

mutation CreateUser($input: ICreateUserInput!) {
  Execute Mutation
  createUser(input: $input) {
    id
  }
}

```

Fonte: Autoria Própria

As operações apresentadas acima exemplificam *queries* e *mutations* reais utilizadas para consultar ou criar dados por meio da API GraphQL. Essas operações podem ser executadas tanto pelo aplicativo móvel quanto pelo painel web ou

diretamente no GraphQL Playground, permitindo testar, validar e consumir os serviços disponibilizados pelo backend de maneira unificada.

4.7 CONSIDERAÇÕES FINAIS

De forma geral, a apresentação da aplicação evidenciou como cada componente — *mobile*, *web* e *backend* — foi estruturado para atender às demandas operacionais do PNCF. As telas do aplicativo móvel demonstraram a organização das fiscalizações em campo, destacando funcionalidades como sincronização de dados, navegação por abas e suporte a operação *offline*, elementos essenciais para o trabalho dos fiscais em regiões com conectividade limitada. O painel *web*, por sua vez, revelou a lógica administrativa do sistema, permitindo o gerenciamento de fiscalizações, assentamentos, UTEs, usuários, além da atribuição de atividades e visualização de formulários enviados pelos fiscais. Já a análise do *backend* mostrou a robustez da API GraphQL, com exemplos do playground, do schema e das operações utilizadas para manter a integridade e a comunicação entre os módulos da aplicação. Em conjunto, essas demonstrações reforçam a coerência da arquitetura implementada e evidenciam que a solução proposta atende de maneira integrada, eficiente e segura aos requisitos definidos ao longo do projeto.

5 CONCLUSÃO

O desenvolvimento do sistema apresentado configura-se como uma resposta tecnológica às limitações identificadas no processo de fiscalização das propriedades adquiridas por meio do PNCF – Terra Brasil. A integração entre as plataformas web e mobile constitui um avanço significativo no acompanhamento das atividades de campo, proporcionando maior eficiência, transparência e aderência às demandas contemporâneas de digitalização e modernização dos serviços públicos.

A adoção de tecnologias atuais, como *NestJS*, *Next.js*, *Flutter* e *Wasabi Cloud Storage*, em conjunto com a linguagem *TypeScript*, resultou na construção de uma arquitetura robusta, escalável e de elevado desempenho. As escolhas tecnológicas realizadas refletem a preocupação em assegurar não apenas a funcionalidade do sistema, mas também sua manutenção, segurança e capacidade de evolução, aspectos fundamentais para soluções utilizadas em ambientes institucionais e governamentais.

Além disso, o projeto contribui diretamente para o aperfeiçoamento da gestão pública, ao oferecer mecanismos que permitem o monitoramento em tempo real das fiscalizações, o armazenamento seguro dos registros obtidos em campo e o acesso facilitado às informações por parte dos gestores das UTEs. Dessa forma, o sistema consolidou-se como uma ferramenta relevante para o aumento da eficiência administrativa e para o fortalecimento da transparência no uso dos recursos destinados à reforma agrária e ao crédito fundiário.

Por fim, espera-se que este trabalho sirva de referência para futuras iniciativas voltadas à aplicação de tecnologias na gestão pública, podendo ser expandido ou adaptado para outros programas governamentais que demandem coleta estruturada, controle e análise de dados em campo. Assim, o sistema desenvolvido reafirma o papel estratégico da tecnologia como instrumento de transformação social e como suporte às políticas públicas direcionadas ao desenvolvimento rural sustentável.

REFERÊNCIAS

IBGE – INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Área territorial brasileira 2021**. Rio de Janeiro: IBGE, 2022. Disponível em: <https://cidades.ibge.gov.br/brasil/panorama>. Acesso em: 31 de maio de 2022.

PONTES, Nádia. **Quem produz os alimentos que chegam à mesa do brasileiro?**. [2017]. Disponível em: <http://www.asbraer.org.br/index.php/rede-de-noticias/item/3510-quem-produz-os-alimentos-que-chegam-a-mesa-do-brasileiro#:~:text=Quando%20se%20consideram%20alimentos%20consumidos,%2C%20milho%2C%20leite%2C%20batata>. Acesso em: 20 abr. 2022

GOV. **Terra Brasil - Programa Nacional de Crédito Fundiário**. [2022]. Disponível em: <https://www.gov.br/agricultura/pt-br/assuntos/agricultura-familiar/credito>. Acesso em: 20 abr. 2022

GOV. **Assentamentos**. [2021]. Disponível em: <https://www.gov.br/incra/pt-br/assuntos/reforma-agraria/assentamentos>. Acesso em: 20 abr. 2022

REFUSA. **O que é o projeto REFUSA?**. [2020]. Disponível em: <http://www.projeto refusa.com/index.html>. Acesso em: 20 abr. 2022

IDAF. **Unidade Técnica Estadual**. [2020]. Disponível em: <https://idaf.es.gov.br/ute-esl>. Acesso em: 26 abr. 2022.

BOOCH, G; RUMBAUGH, J e JACOBSON, I: **UML, Guia do Usuário**: tradução; Fábio Freitas da Silva, Rio de Janeiro, Campus , 2012. Acesso em: 06 jun. 2022.

Berners-Lee T.; Fielding R.; Frystyk H. **Hypertext Transfer Protocol -- HTTP/1.0** (em inglês). Internet Engineering Task Force. Disponível em: <https://www.w3.org/Protocols/HTTP/1.0/spec.html> Acesso em: 07 de junho de 2022.

Harada, E. **O que é Figma e como você pode usufruir dessa ferramenta de design**. [2022]. Disponível em: <https://www.tecmundo.com.br/software/236320-figma-voce-usufruir-dessa-ferramenta-design.htm>>. Acesso em: 08 jun. 2022.

TYPESCRIPT. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2021. Disponível em: <https://pt.wikipedia.org/w/index.php?title=TypeScript&oldid=61673233>. Acesso em: 27 abr. 2022.

NODE.JS. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2021. Disponível em: <https://pt.wikipedia.org/w/index.php?title=Node.js&oldid=61833518>. Acesso em: 27 abr. 2022.

GRAPHQL. In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2017. Disponível em: <https://pt.wikipedia.org/w/index.php?title=GraphQL&oldid=49315253>. Acesso em: 29 abr. 2022.

REACT (JAVASCRIPT). In: WIKIPÉDIA, a enciclopédia livre. Flórida: Wikimedia Foundation, 2022. Disponível em: [https://pt.wikipedia.org/w/index.php?title=React_\(JavaScript\)&oldid=63480980](https://pt.wikipedia.org/w/index.php?title=React_(JavaScript)&oldid=63480980). Acesso em: 28 mai. 2022.

GOOGLE. **Cloud Storage Documentation.** Mountain View: Google LLC, 2024. Disponível em: <https://cloud.google.com/storage>. Acesso em: 08 out. 2025.

MICROSOFT. **Azure Blob Storage Overview.** Redmond: Microsoft Corporation, 2024. Disponível em: <https://learn.microsoft.com/azure/storage/blobs>. Acesso em: 08 out. 2025.

WASABI TECHNOLOGIES. **Wasabi Hot Cloud Storage Documentation.** Boston: Wasabi Technologies, 2024. Disponível em: <https://wasabi.com/resources/documentation>. Acesso em: 08 out. 2025.

AHMAD, Shafaq; WASIM, Muhammad. **Agile Methodologies in Modern Software Development.** *Journal of Software Engineering*, v. 17, n. 2, 2023. Acesso em: 15 out. 2025.

EFENDI, M.; MUTAWALLI, M.; AKBAR, R. **Scrum Implementation for Efficient Software Development.** *International Journal of Advanced Computer Science*, v. 15, n. 4, 2024. Acesso em: 15 out. 2025.

GHANDI, Samira et al. **Scrum in Industry: Challenges and Adaptations.** *IEEE Software*, v. 34, n. 2, 2017. Acesso em: 15 out. 2025.

GUPTA, A. et al. **Adaptability of Agile Processes in Dynamic Environments.** *Software Practice & Experience*, v. 52, n. 1, 2022. Acesso em: 18 out. 2025.

HEIBGES, Philipp; JUNGNICHEL, Katharina; FEUFEL, Markus. **User-centered Review Processes in Agile Teams.** *ACM Transactions on Software Engineering*, v. 49, n. 3, 2023. Acesso em: 18 out. 2025.

PRESSMAN, Roger S.; MAXIM, Bruce. **Engenharia de Software: uma abordagem profissional**. 9. ed. São Paulo: McGraw-Hill, 2020.
Acesso em: 30 out. 2025.

SCHWABER, Ken; SUTHERLAND, Jeff. **The Scrum Guide: The Definitive Guide to Scrum**. Scrum.org, 2020.
Acesso em: 31 out. 2025.

SOMMERVILLE, Ian. **Engenharia de Software**. 10. ed. São Paulo: Pearson, 2019.
Acesso em: 31 out. 2025.

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **The Unified Modeling Language User Guide**. 2. ed. Boston: Addison-Wesley, 2005.
Acesso em: 01 nov. 2025.

FOWLER, Martin. **UML Essencial: Um Breve Guia para a Linguagem-Padrão de Modelagem de Objetos**. 3. ed. Porto Alegre: Bookman, 2004.
Acesso em: 01 nov. 2025

APÊNDICE A - LEVANTAMENTO DE REQUISITOS

1 INTRODUÇÃO

Este apêndice apresenta o levantamento de requisitos do sistema proposto, descrevendo de forma objetiva as necessidades do software e os principais casos de uso, a fim de apoiar o entendimento e o desenvolvimento da solução.

1.1 Objetivo

O objetivo desse documento é descrever os requisitos do *software* proposto, bem como facilitar o entendimento dos casos de uso.

1.2 Escopo

Desenvolver um *software* de fiscalização e gestão, para ajudar o governo brasileiro a supervisionar o crédito distribuído via Programa Nacional de Crédito Fundiário.

1.3 Visão Geral

O sistema permitirá aos gestores das UTEs e administradores, controlarem e supervisionarem todas as etapas das fiscalizações realizadas pelos fiscais, que por sua vez poderão fiscalizar os assentamentos tanto de maneira *online* quanto *offline*.

2 DESCRIÇÃO

A descrição dos casos de uso tem como finalidade apoiar o levantamento e a validação dos requisitos, além de servir como referência para as etapas de desenvolvimento, testes e manutenção do sistema.

2.1 Atores

ID	Nome	Descrição
A-01	Administrador	Principal usuário do sistema, responsável por manter a integridade dos dados e do sistema.
A-02	Gestor de UTE	Responsável pela gestão das terras do seu estado, e também acompanhar as fiscalizações feitas pelos

		fiscais.
A-03	Fiscal	Usuário responsável por fazer as fiscalizações nos assentamentos requisitados.

2.2 Funções do sistema

ID	Nome	Descrição
F-01	Acessar sistema	Os usuário poderão acessar o sistema para ter acesso às funções do sistema
F-02	Cadastrar Usuários	Os administradores e gestores de UTE poderão cadastrar novos usuário para ele terem acesso ao sistema
F-03	Cadastrar Unidade Técnicas Estaduais (UTE)	O sistema deve permitir o cadastro de Unidade Técnicas Estaduais (UTE), para que elas gerenciem os assentamentos de um estado
F-04	Cadastrar Assentamentos	O sistema deve permitir a criação de assentamentos e a vinculação do mesmo a uma UTE, para possibilitar a gestão das fiscalizações daquele assentamento.
F-05	Incluir Agentes Fiscais em uma fiscalização	O sistema deve permitir a inclusão de usuário com perfil fiscal em fiscalizações, para que os mesmo possa realizar as fiscalizações.

F-06	Incluir Parcelas em batch(lote) nos Assentamentos	Os administradores e gestores de UTE poderão incluir várias parcelas de uma só vez.
F-07	Listar UTEs	O sistema deve permitir a listagem de assentamentos para o gerenciamento dos seus assentamentos.
F-08	Listar Assentamentos	O sistema deve permitir a listagem de assentamentos para o gerenciamento da suas parcelas
F-09	Listar Fiscais	O sistema deve permitir o cadastro de Unidade Técnicas Estaduais (UTE), para que elas gerenciem os assentamentos e usuários vinculados a ela.
F-10	Criar fiscalização	O sistema deve permitir a inclusão de lotes em batch, ou seja vários lote de uma vez.
F-11	Realizar fiscalização	O sistema deve permitir que os fiscais possam realizar fiscalizações nos assentamentos tanto de maneira <i>online quanto offline</i> .
F-12	Recuperar acesso	O sistema deve permitir que todos os usuários possam fazer a troca da sua senha.

2.3 Casos de uso

C-04	Cadastrar Usuários
Ator Principal	Administrador, Gestor de UTE.
Descrição	Para cadastrar novos usuários, o administrador ou gestor deverá preencher todos os dados.
Propósito	Dar acesso ao sistema a outros usuário
Pré-Condição	O usuário que será cadastrado deverá ser gestor de uma UTE ou um fiscal.
Pós-Condição	
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de cadastro de usuários; 5. Preencher o formulário; 6. Confirmar o formulário de cadastro do usuário;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de cadastro de usuários; 5. Preencher o formulário; 6. Confirmar o formulário de cadastro do usuário; 7. O usuário já possui cadastro no sistema
Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>;

	3. Utilizar o sistema; 4. Preencher o formulário; 5. Confirmar o formulário de cadastro do usuário; 6. O administrador digitou algum campo errado ou esqueceu de preencher algum campo obrigatório;
--	--

C-05	Acessar sistema
Ator Principal	Administrador, Gestor de UTE, fiscal.
Descrição	Os usuários deverão preencher o formulário para realizar o <i>login</i> na plataforma.
Propósito	Ter acesso ao sistema.
Pré-Condição	O usuário deve estar cadastrado.
Pós-Condição	
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. Utilizar o sistema;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. E-mail não localizado ou senha incorreta;
Fluxo Exceção	1. Preencher o formulário; 2. Não foi possível estabelecer conexão com o sistema;

	3. Mostrar o erro ao usuário;
--	-------------------------------

C-06	Cadastrar Assentamentos
Ator Principal	Administrador, Gestor de UTE.
Descrição	O usuário administrador ou gestor de UTE deverá preencher o formulário para o cadastro de assentamentos.
Propósito	Gerenciar as fiscalizações daquele assentamento.
Pré-Condição	Ter uma UTE cadastrada naquele estado.
Pós-Condição	Retornar o assentamento cadastrado.
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de Assentamentos; 5. Preencher o formulário; 6. Confirmar o formulário de cadastro do assentamento;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de Assentamentos; 5. Preencher o formulário; 6. Confirmar o formulário de cadastro

	do assentamento; 7. Assentamento já cadastrado;
Fluxo Exceção	1. Preencher o formulário; 2. Não foi possível estabelecer conexão com o sistema; 3. Mostrar o erro ao usuário;

C-07	Incluir Parcelas em batch(lote) nos Assentamentos
Ator Principal	Administrador, Gestor de UTE.
Descrição	O usuário administrador ou gestor de UTE deverá preencher o formulário para incluir várias parcelas de uma vez em um assentamento.
Propósito	Gerenciar as fiscalizações daquele assentamento.
Pré-Condição	Ter um assentamento cadastrado
Pós-Condição	
Fluxo Principal	1. Preencher o formulário; 2. Confirmar o formulário de cadastro de parcelas;
Fluxo Alternativo	1. Preencher o formulário; 2. Confirmar o formulário de cadastro do assentamento;

	3. Parcelas já cadastradas;
Fluxo Exceção	1. Preencher o formulário; 2. Não foi possível estabelecer conexão com o sistema; 3. Mostrar o erro ao usuário;

C-08	Listar UTEs
Ator Principal	Administrador
Descrição	O usuário administrador poderá listar todas as UTEs.
Propósito	Gerenciar as UTEs.
Pré-Condição	Ter acesso ao sistema.
Pós-Condição	Deve ser mostrado um campo de busca onde será possível realizar a procura de UTEs incluindo filtros.
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. Utilizar o sistema; 4. Acessar a aba de UTEs;
Fluxo Alternativo	

Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de UTEs; 5. Não foi possível estabelecer conexão com o sistema; 6. Mostrar o erro ao usuário;
----------------------	--

C-09	Listar Assentamentos
Ator Principal	Administrador, Gestor de UTE, fiscal.
Descrição	O usuário administrador poderá listar todas os assentamentos, para poder cadastrar novos ou incluir fiscais nas fiscalizações daquele assentamento. O usuário gestor só poderá listar os assentamentos da sua UTE. E os fiscais só poderão listar os assentamentos que eles estão vinculados.
Propósito	Gerenciar os assentamentos.
Pré-Condição	Ter acesso ao sistema.
Pós-Condição	Deve ser mostrado um campo de busca onde será possível realizar a procura de assentamentos incluindo filtros.
Fluxo Principal	1. Preencher campo de email e senha;

	2. Efetuar <i>login</i> ; 3. Utilizar o sistema; 4. Acessar a aba de Assentamentos;
Fluxo Alternativo	
Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i> ; 3. Utilizar o sistema; 4. Acessar a aba de Assentamentos; 5. Não foi possível estabelecer conexão com o sistema; 6. Mostrar o erro ao usuário;

C-10	Listar Fiscais
Ator Principal	Administrador, Gestor de UTE.
Descrição	O usuário administrador poderá listar todas os fiscais, e o usuário gestor poderá listar os fiscais vinculados a sua UTE.
Propósito	Gerenciar os fiscais.
Pré-Condição	Ter acesso ao sistema.
Pós-Condição	Deve ser mostrado um campo de busca onde será possível realizar a procura de fiscais incluindo

	filtros.
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de fiscais;
Fluxo Alternativo	
Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de fiscais; 5. Não foi possível estabelecer conexão com o sistema; 6. Mostrar o erro ao usuário;

C-11	Criar fiscalização
Ator Principal	Administrador, Gestor de UTE.
Descrição	O administrador ou gestor de UTE, poderão criar fiscalizações extraordinárias ou por amostragem.
Propósito	Verificar se os assentamentos estão cumprindo os requisitos do Programa Nacional de Crédito Fundiário ou se estão com alguma irregularidade.
Pré-Condição	Ter acesso ao sistema.

Pós-Condição	
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de assentamento; 5. Selecionar um assentamento; 6. preencher o formulário para iniciar uma fiscalização;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de assentamento; 5. Selecionar um assentamento; 6. Já existe uma fiscalização em andamento naquele assentamento;
Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de assentamento; 5. Selecionar um assentamento; 6. Não foi possível estabelecer conexão com o sistema; 7. Mostrar o erro ao usuário;

C-12	Realizar fiscalização
Ator Principal	Fiscal.
Descrição	O fiscal poderá realizar uma fiscalização em um determinado assentamento.

Propósito	Verificar se os assentamentos estão cumprindo os requisitos do Programa Nacional de Crédito Fundiário ou se estão com alguma irregularidade.
Pré-Condição	Ter acesso ao sistema.
Pós-Condição	
Fluxo Principal	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de fiscalizações; 5. Selecionar uma fiscalização;
Fluxo Alternativo	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de fiscalizações; 5. Selecionar uma fiscalização; 6. O fiscal já está fiscalizando aquele assentamento;
Fluxo Exceção	1. Preencher campo de email e senha; 2. Efetuar <i>login</i>; 3. Utilizar o sistema; 4. Acessar a aba de assentamento; 5. Selecionar um assentamento; 6. Não foi possível estabelecer conexão com o sistema; 7. Mostrar o erro ao usuário;

C-13	Recuperar acesso
Ator Principal	Administrador, Gestor de UTE, Fiscal.
Descrição	Os usuários poderão trocar a sua senha.
Propósito	Recuperar o acesso caso o usuário tenha perdido ou esquecido sua senha.
Pré-Condição	Ter cadastro no sistema.
Pós-Condição	o usuário deverá seguir os passos enviado no e-mail cadastrado
Fluxo Principal	1. Acessar a página de recuperação de senha; 2. preencher o campo de email; 3. Receber o e-mail de recuperação 4. O usuário deverá acessar o link recebido via e-mail. 5. preencher o campo com a confirmação da nova senha;
Fluxo Alternativo	1. Acessar a página de recuperação de senha; 2. preencher o campo de email; 3. O e-mail não foi recebido;
Fluxo Exceção	1. Acessar a página de recuperação de senha; 2. preencher o campo de email; 3. Receber o e-mail de recuperação 4. O usuário deverá acessar o link recebido via e-mail. 5. preencher o campo com a

	confirmação da nova senha; 6. A nova senha não atende aos padrões de segurança.
--	---

2.4 Diagrama de casos de uso

Diagrama de Caso de uso do administrador

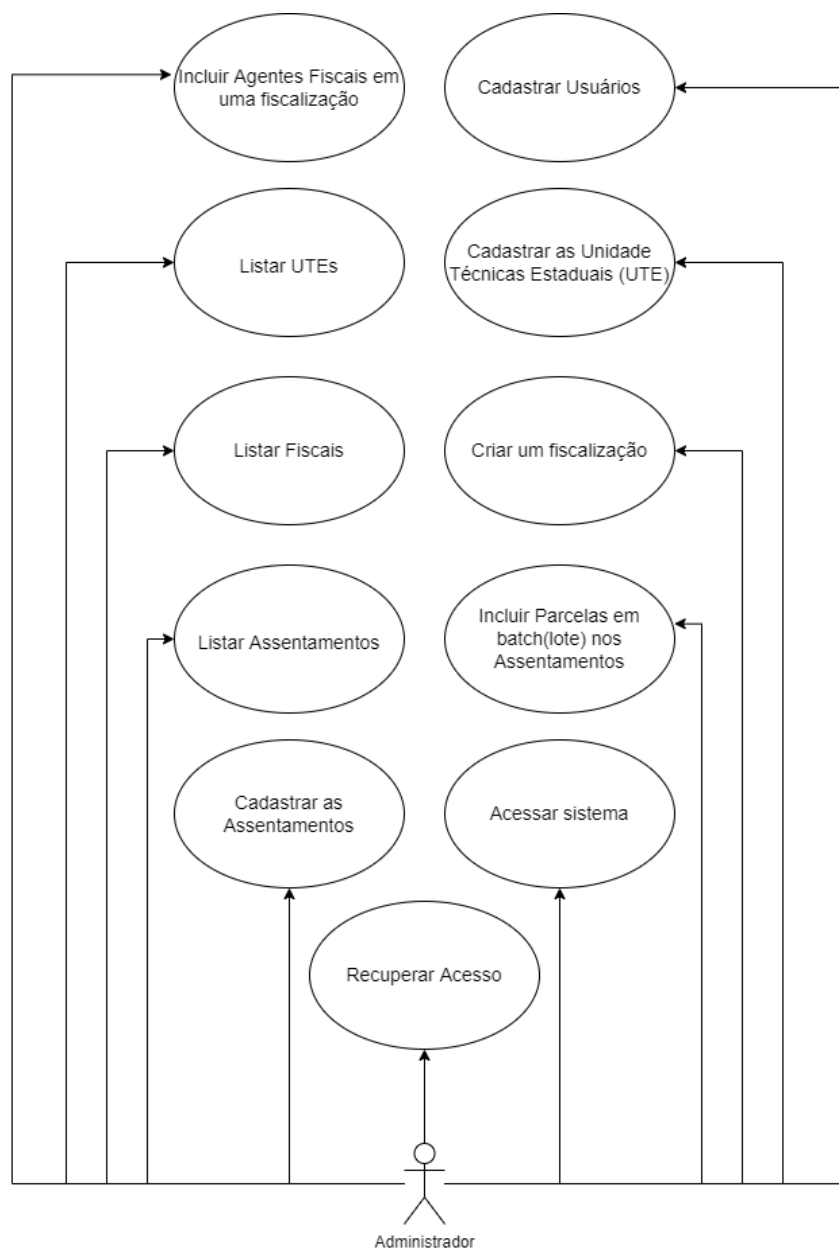


Diagrama de caso de uso do gestor de UTE

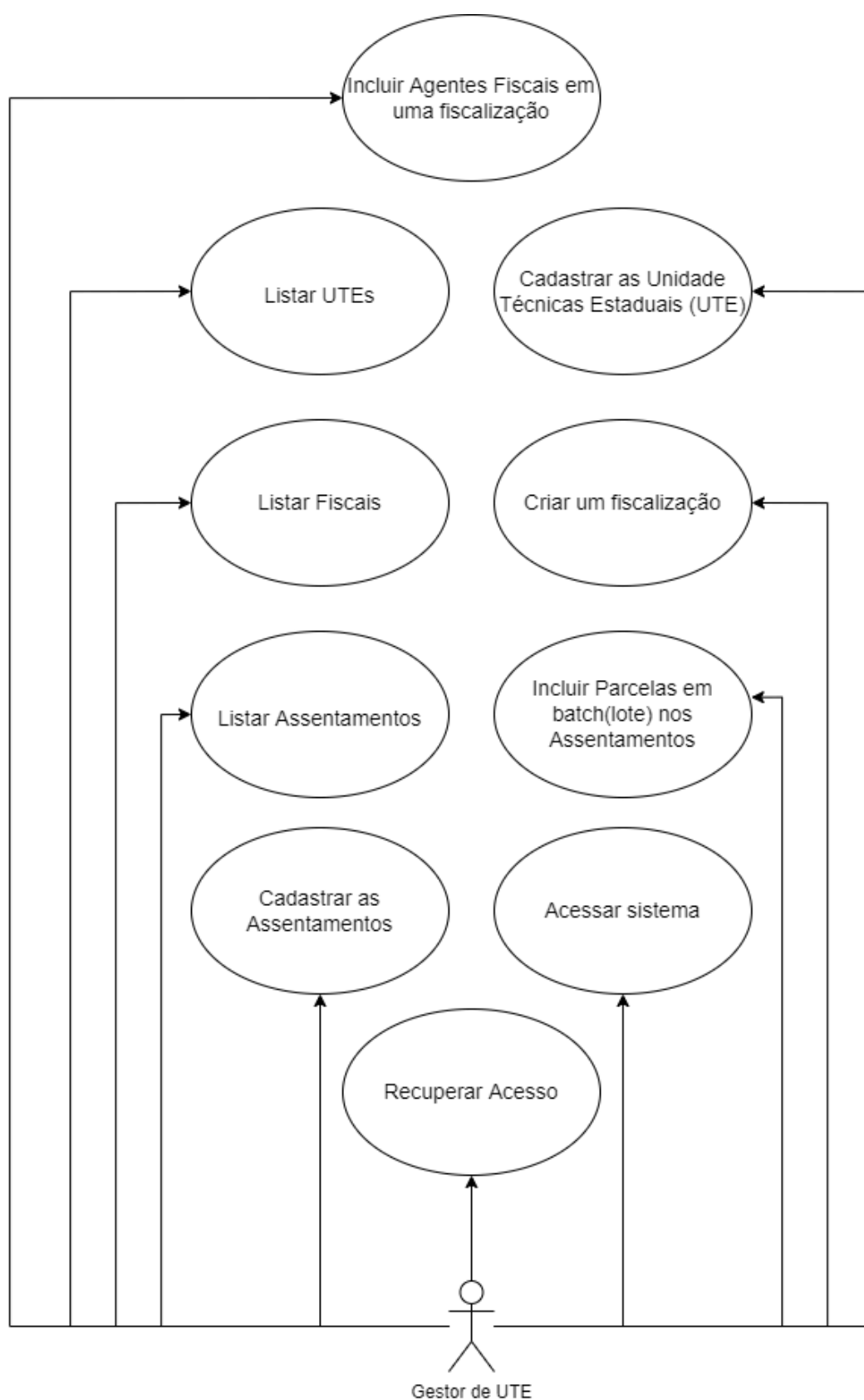
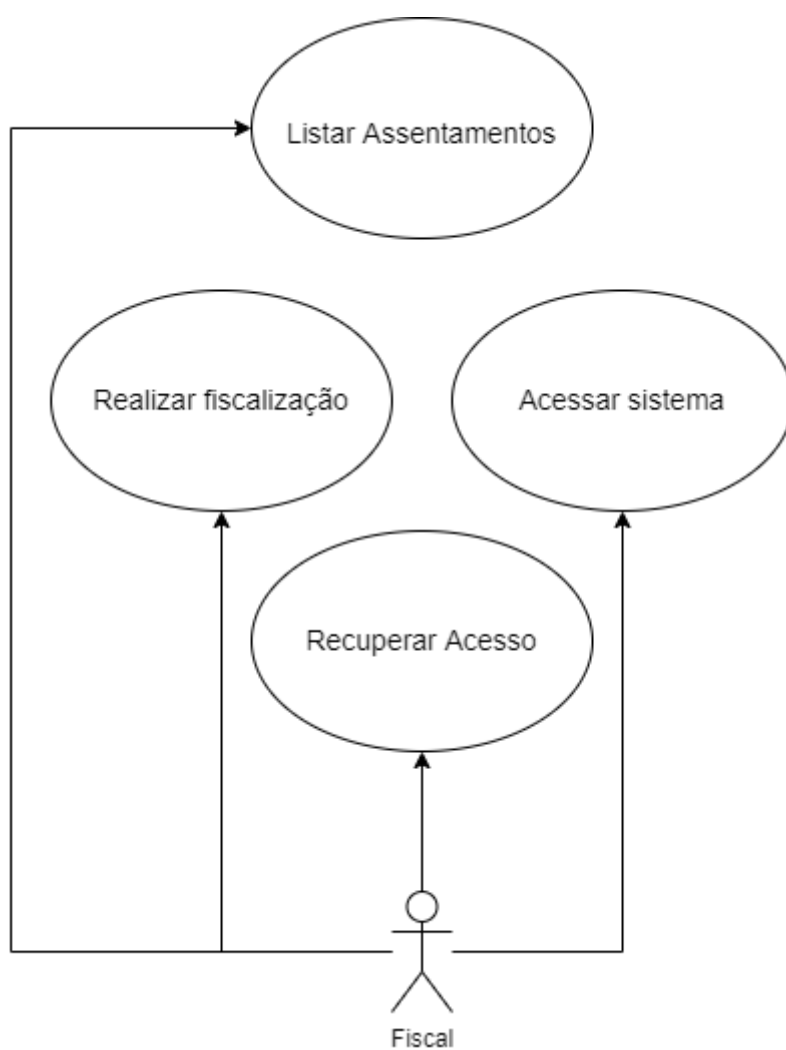


Diagrama de caso de uso do fiscal



ANEXO A - INTERFACES DO PROJETO

1 WEB

Tela de login



A screenshot of the login page for Terra Brasil. The page has a light gray background. In the center, there is a white card with a shadow. At the top of the card is the Terra Brasil logo, which consists of a stylized green and yellow sun rising over green hills, with the text "TERRA BRASIL" below it, followed by "Programa Nacional de Crédito Fundiário". Below the logo are two input fields: the first is labeled "E-mail" with an envelope icon, and the second is labeled "Senha" with a key icon and a toggle eye icon. Below the password field is a blue link that says "Esqueceu sua senha?". At the bottom of the card is a blue button with the text "Entrar".

MINISTÉRIO DA
AGRICULTURA, PECUÁRIA
E ABASTECIMENTO

 **PÁTRIA AMADA
BRASIL**
GOVERNO FEDERAL

Recuperação de senha



Digite o e-mail cadastrado para o envio das instruções de recuperação de senha.

 E-mail

Enviar

ou

[Voltar para fazer login](#)

2 Mobile

Tela de Login



TERRA BRASIL
Programa Nacional de
Crédito Fundiário

Login

[Esqueceu a senha?](#)

Entrar

MINISTÉRIO DA
AGRICULTURA, PECUÁRIA
E ABASTECIMENTO



**PÁTRIA AMADA
BRASIL**
GOVERNO FEDERAL

Recuperação de senha



Recuperação de Senha

Digite o e-mail cadastrado para o envio das instruções de recuperação de senha.

Enviar

OU

[Voltar para fazer login](#)

MINISTÉRIO DA
AGRICULTURA, PECUÁRIA
E ABASTECIMENTO



Aba de Informações sobre a renda do ocupante

15:10


Criação de Animais


Equipamentos e Máquinas


Rendas


Força de Trabalho


Assistência Técnica ATER

\$ Rendas

\$ Renda Agropecuária Bruta Anual

Renda Agricultura Mensal R\$

Renda Criação/Venda de Animais Mensal R\$

Renda Venda do Leite Mensal R\$

Renda Agropecuária Bruta Anual Total Obtida no Lote R\$: 1500 reais

\$ Renda Não Agrícola Bruta Familiar Anual

Aposentadoria/Pensão Mensal R\$

Trabalho Fora do Lote Mensal R\$

Comércio , Venda ou Serviços Mensais R\$

Benefícios

BPC 

Bolsa Família

+ Adicionar Benefício

Valor Total dos Benefícios R\$

Renda Não Agrícola Bruta Anual Total Obtida R\$: 1500 reais

Cancelar

 Salvar



Aba de Informações sobre o responsável técnico

15:10


Operação de
Financiamento


Dados dos
Beneficiários


Tipificação e
Ocupantes Irregulares


Infraestruturas do
Lote


Exploração I

Responsável Técnico - ATER

Contrato de ATER Vigente?

Não

Sim

Entidade Prestadora

Técnico Responsável

Cancelar

 Salvar

III

O

<

Aba de Informações sobre Constatações ambientais

15:10


Áreas de
Reservação


Constatações
Ambientais


Imagens do Lote


Outras
Informações


Assinaturas


Termos de
Ciência



Constatações Ambientais Irregulares

Existe no Lote/Parcela?

Não

Sim

Uso inadequado das terras em relação a sua aptidão?

Não

Sim

Erosão e Compactação do solo?

Não

Sim

Uso de queimadas e agrotóxicos sem controle?

Não

Sim

Práticas inadequadas de calagem e adubação?

Não

Sim

Outras Informações e Observações:

digite as observações aqui

Cancelar

 Salvar

III

O

<

Aba de Informações sobre a exploração do lote

15:10

Criação de Animais

Equipamentos e Máquinas

Rendas

Força de Trabalho

Assistência Técnica ATER

Exploração Agrícola do Lote

Exploração Efetiva?

Não

Sim

Espécie Cultivada

Unidade

Quantidade

Local de Cultivo

Tipo de cultura:

Lote Agropecuário

Agricultura

Horticultura

Fruticultura

+ Adicionar Item

Itens Cadastrados

Lote Agropecuário

Agricultura

Arroz

500 KG

✕

Soja

500 KG

✕

Horticultura

Cenoura

500 KG

✕

Fruticultura

Laranja

500 KG

✕

Lote Agrovila

Agricultura

Arroz

500 KG

✕

Soja

500 KG

✕

Horticultura

Cenoura

500 KG

✕

Fruticultura

Laranja

500 KG

✕

Área Coletiva

Agricultura

Arroz

500 KG

✕

Soja

500 KG

✕

Horticultura

Cenoura

500 KG

✕

Fruticultura

Laranja

500 KG

✕

Cancelar

Salvar

Aba de Informações sobre criação de animais

15:10


Criação de Animais


Equipamentos e Máquinas


Rendas


Força de Trabalho


Assistência Técnica ATER

Criação de Animais

Animais Cadastrados

	Quantidade
 Galinhas	2 
 Patos	2 
 Bovinos	2 
 Perus	2 
 Equinos	2 
 Peixes	2 
 Espécie Peixes Quantidade 45  Adicionar	

Produção Nominal

Produção Nominal de Ovos por Dia

Preço da Venda da Dúzia de Ovos em R\$

Produção Nominal de Leite em Litros por Dia

Preço da Venda do Leite em R\$ por Litro

Cancelar

 Salvar

Aba de Informações sobre equipamentos e máquinas

15:10


Criação de Animais


Equipamentos e Máquinas


Rendas


Força de Trabalho


Assistência Técnica ATER

Uso de Equipamentos e Máquinas

Digite aqui outro equipamento ou máquina

+ Adicionar

Equipamentos de Uso Humano / Manual

Enxada

- 1 +

Foice

- 1 +

Martelo

- 1 +

Equipamentos de Maquinário / Máquinas

Trator

- 1 +

Colheitadeira

- 1 +

Semeadora

- 1 +

Cancelar

 Salvar

III

O

<

Aba de Informações sobre força de trabalho do lote

15:10





Criação de Animais



Equipamentos e Máquinas



Rendas



Força de Trabalho



Assistência Técnica ATER



Força de Trabalho



Membros da Família

Quantidade de residentes no lote

Quantidade que trabalham no lote

Quantidade que trabalham fora do lote



Mão de Obra Assalariada

Quantidade de funcionários temporários

Quantidade de funcionários em empreitas

Quantidade de funcionários permanentes

Cancelar

Salvar

III

O

<

Aba de Informações sobre a preservação

15:10


Operação de Financiamento


Dados dos Beneficiários


Tipificação e Ocupantes Irregulares


Infraestruturas do Lote


Exploração

 **Preservação**

 **Área de Reserva Legal - ARL**

Existe Área de Reserva Legal no lote?

Não

Sim

Foi observada alguma alteração?

Não

Sim

Há necessidade de recomposição/compensação?

Não

Sim

 **Área de Preservação Permanente - APP**

Existe APP no lote?

Não

Sim

Foi observada alguma alteração?

Não

Sim

Há necessidade de recomposição/compensação?

Não

Sim

Cancelar

 Salvar

III

O

<

Aba de assinaturas

15:10

Áreas de Preservação Assinaturas Imagens do Lote Outras Informações Assinaturas Termos de Ciência

Assinaturas

Beneficiário

Incididunt ut labore et dolore		✕
Incididunt ut labore et dolore		✕

Ocupante

Incididunt ut labore et dolore		✕
Incididunt ut labore et dolore		✕

Assinatura do Beneficiário, Titular 1, 2 ou outro:

 Limpar

 Editar

+ Adicionar assinatura

Outras Observações

Outras Informações e Observações:

digite as observações aqui

Fiscal

Incididunt ut labore et dolore		✕
--------------------------------	--	---

Cancelar Salvar

ANEXO B - FORMULÁRIO DE FISCALIZAÇÃO

	Timbre - Governo do Estado...UF Endereço; Telefone; e-mail UNIDADE ESTADUAL...		
	FORMULÁRIO DE FISCALIZAÇÃO OCUPACIONAL		
1 - IDENTIFICAÇÃO DA OPERAÇÃO DE FINANCIAMENTO			
Número Único da Operação OBTER/CRÉDITO:	Proposta SIG/CF:	Município / UF:	Nº de Famílias
Processo Administrativo Nº:	Linha de Financiamento:	Data da Contratação:	Agente Financeiro
Identificação do Imóvel:	LOTE nº:	Área do Lote (ha):	
Projeto Coletivo:			CNPJ:
REGULARIDADE OCUPACIONAL			
2 - DADOS DO BENEFICIÁRIO			
Titular 01	CPF:	RG:	Número de Filhos:
Titular 02	CPF:	RG:	Outros Dependentes:
Endereço de Correspondência:			
Contatos: Telefone ()		E-mail:	
Observações:			
3 - TIPIFICAÇÃO DA OCUPAÇÃO DA PARCELA			
☐ Lote/Parcela Ocupada REGULARMENTE		☐ Substituição com anuência	
☐ Lote/Parcela Ocupada IRREGULARMENTE		☐ Em processo de Assunção - Substituição	
Data da Ocupação:			
☐ Lote/Parcela Vaga Abandono - Tempo de Abandono:			
☐ Compra/Venda - Data: Valor: R\$ () Arrendamento			
☐ Permuta - C/ Lote/Parcela: () Preposto/procuração			
☐ Reconcentração Fundiária - Número de Lotes n.º			
4 - DADOS DO OCUPANTE IRREGULAR			
Nome do Ocupante		CPF:	RG:
Nome do Cônjuge		CPF:	RG:
Cadastro no CADÚnico: () Sim () Não		Cadastro no Programa Bolsa Família: () Sim () Não	
Endereço de Correspondência:			
Contatos: Telefone ()		E-mail:	
Outras informações:			
5 - INFRA-ESTRUTURA DO LOTE (agropecuário)			
Acessou PRONAF: () Sim () Não		DAP/CAF Válida () Sim () Não	
Conectividade INTERNET: () Sim () Não		Acessou: () PAA () PNAE () OUTRAS:	
Habituação Existente: () Sim () Não		Habituação Ocupada: () Sim () Não	
☐ Casa de Alvenaria () Casa de Madeira () Casa de Taipa () Rede Elétrica () Poço () Cisterna () Mina/Nascente () Açude () Irrigação () Aviário			
☐ Pociça () Curral () Aprisco () Silo () Cerca Perimetral do Lote () Cerca na Reserva () Outros () Outros:			
6 - INFRA-ESTRUTURA URBANA (agrovila)			
Conectividade INTERNET: () Sim () Não		Habituação n.º Situação: () Abandonada () Ocupada () Alugada	
Tipologia: () Alvenaria () Madeira		() Energia Elétrica () Água Encanada () Poço () Cisterna () Unidade Sanitária () Fossa Séptica () Rede de Esgoto () Outros	
7 - EXPLORAÇÃO EFETIVA			
Lote/Parcela Explorada		() Sim () Não	
Cultura Explorada			
Agricultura		Horticultura	
Espécies Cultivadas	Unid.	Quant.	Fruticultura
Espécies Cultivadas	Unid.	Quant.	Espécies Cultivadas
Espécies Cultivadas	Unid.	Quant.	Espécies Cultivadas
7.1 - EXPLORAÇÃO EFETIVA (Lote Agropecuário)			
7.2 - EXPLORAÇÃO EFETIVA (Lote - Agrovila)			
7.3 - EXPLORAÇÃO EFETIVA (Área Coletiva)			

8 - CRIAÇÃO DE ANIMAIS					
Especificação	Nº Cabeças	Especificação	Nº Cabeças	Especificação	Nº Cabeças
Galinhas/Frango		Bovinos		Caprinos	
Patos/Perus		Equinos		Ovinos	
Peixes		Muare		Suínos	
Outros:					
Produção Nominal de Ovos por Dia:			Preço da Venda da Dúzia de Ovos em R\$:		
Produção Nominal de Leite em Litros por Dia:			Preço da Venda do Leite em R\$ por Litro:		

9 - EQUIPAMENTOS E MÁQUINAS
() Trator () Grade () Arado () Sulcador () Forrageira () Semeadeira () Motocultivar/Tratorito () Pulverizador Costal () Plantadeira/Adubadora
() Carro de Boi () Arado Animal () Carroça Animal () Enxadas () Foice () Picaretas () Facão () Outros:

10 - RENDA AGROPECUÁRIA BRUTA ANUAL		
Renda Agricultura Mensal R\$:	Renda Criação/Venda de Animais Mensal R\$:	Renda Venda do Leite Mensal R\$:

Renda Agropecuária Bruta (ANUAL-TOTAL) Obtida no Lote em R\$:

11 - RENDA NÃO AGRÍCOLA BRUTA FAMILIAR ANUAL		
Aposentadoria/Pensão Mensal R\$:	Trabalho Fora do Lote Mensal R\$:	Comércio-Venda-Serviços Mensal R\$:

Programas Sociais: () Bolsa Família () BPC () Outros

Valor Total dos Benefícios R\$:

Renda Não Agrícola Bruta Familiar (ANUAL-TOTAL) Obtida em R\$:

12 - FORÇA DE TRABALHO	
Mão – de – Obra Lote/Parcela	Número de pessoas
Membros da Família	
Trabalham no Lote	
Trabalham fora do Lote	
Residentes no Lote	
Mão-de-obra Assalariada	
Temporária no Lote	
Empreitas no Lote	
Permanentes no Lote	

13 - ASSISTÊNCIA TÉCNICA - ATER	
Contrato de ATER vigente: () Sim () Não	
Entidade Prestadora:	Técnico Responsável:
Obs.:	

14 - ÁREA DE RESERVA LEGAL - ARL	
Existe Área de Reserva Legal no lote/parcela? ☐ Não ☐ Sim	Observações:
Foi observada alguma alteração? ☐ Não ☐ Sim	
Há necessidade de recomposição /compensação ☐ Não ☐ Sim	

15 - ÁREA DE PRESERVAÇÃO PERMANENTE - APP	
Existe APP no lote/parcela? ☐ Não ☐ Sim	Observações:
Foi observada alguma alteração? ☐ Não ☐ Sim	
Há necessidade de recomposição /compensação? ☐ Não ☐ Sim	

16 - CONSTATAÇÕES AMBIENTAIS	
Encontra-se no Lote/parcela:	Observações:
Erosão e compactação do solo? ☐ Não ☐ Sim	
Práticas inadequadas de calagem e adubação? ☐ Não ☐ Sim	
Uso de queimadas e agrotóxicos sem controle? ☐ Não ☐ Sim	
Uso inadequado das terras em relação a sua aptidão? ☐ Não ☐ Sim	

Outras Informações e Observações:

Data: ____/____/____ (Beneficiário) Titular 1 ou 2 – Outro (Técnico) Carimbo e Assinatura

Fonte: Manual de fiscalização do PNCF