

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

BRUNO JOSÉ DOS SANTOS WOGT

TÁ SAFE: SISTEMA PARA RASTREABILIDADE DE CELULARES

GUARAPUAVA

2024

BRUNO JOSÉ DOS SANTOS WOGT

TÁ SAFE: SISTEMA PARA RASTREABILIDADE DE CELULARES

Tá Safe: System for Cell Phone Traceability

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Tecnologia em Sistemas para Internet do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Andres Jessé Porfirio

GUARAPUAVA

2024



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

BRUNO JOSÉ DOS SANTOS WOGT

TÁ SAFE: SISTEMA PARA RASTREABILIDADE DE CELULARES

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Tecnologia em Sistemas para Internet do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Data de aprovação: 03/julho/2024

Prof. Andres Jessé Porfirio
Doutor
Universidade Tecnológica Federal do Paraná - Campus Guarapuava

Prof. Diego Marczal
Doutor
Universidade Tecnológica Federal do Paraná - Campus Guarapuava

Prof. Dênis Lucas Silva
Mestre
Universidade Tecnológica Federal do Paraná - Campus Guarapuava

GUARAPUAVA
2024

Dedico este trabalho à minha mãe, Solange,
por todo o amor, apoio e incentivo.

Seu futuro é o que você fizer dele. Então faça-o
bem.
(Dr. Emmett Brown, De Volta para o Futuro III)

RESUMO

Com o lançamento periódico de novos modelos de celulares, os usuários acabam realizando a troca de seus dispositivos com maior frequência se comparado ao passado. Desse modo, a oferta desses aparelhos no mercado de seminovos cresce devido à quantidade de novos anúncios criados diariamente em plataformas de vendas e redes sociais. O furto e roubo desses aparelhos acompanham o crescimento do setor, chegando ao ponto de os usuários que buscam por aparelhos de segunda mão terem dificuldade em definir a legitimidade do dispositivo. Para sanar possíveis problemas ao comprar um celular sem a devida verificação, o presente projeto visa o desenvolvimento de um sistema que ofereça aos seus usuários recursos de rastreabilidade de dispositivos para identificação das etapas que esse aparelho percorreu até ser ofertado no mercado de usados. Com isso, demonstrar que o aparelho foi obtido originalmente em uma transação que gerou uma Nota Fiscal Eletrônica (NF-e) e está vinculado ao Cadastro de Pessoas Físicas (CPF) do primeiro dono. Posteriormente à compra, o registro é transferido para o novo proprietário, criando-se, assim, um novo elo na rastreabilidade desse aparelho.

Palavras-chave: sistema; rastreabilidade; celular.

ABSTRACT

With the periodic release of new cell phone models, users are replacing their devices more frequently compared to the past. Thus, the supply of these devices in the second-hand market grows due to the number of new ads created daily on sales platforms and social networks. The theft and robbery of these devices follow the growth of the sector, to the point that users looking for second-hand devices have difficulty defining the legitimacy of the device. To solve possible problems when buying a cell phone without proper verification, this project aims to develop a system that offers its users device traceability resources to identify the steps that this device has gone through until it is offered in the used market. With this, demonstrate that the device was originally obtained in a transaction that generated an NF-e and is linked to the CPF of the first owner. After the purchase, the registration is transferred to the new owner, thus creating a new link in the traceability of this device.

Keywords: system; traceability; cellphone.

LISTA DE FIGURAS

Figura 1 – Tela inicial Bike Registrada	13
Figura 2 – Modelo de consulta Olho no Carro	14
Figura 3 – Consulta de <i>International Mobile Equipment Identity</i> (IMEI) Celular Legal	15
Figura 4 – Tela de gerenciamento de aparelhos na versão web do Celular Seguro BR.	15
Figura 5 – Ciclo de vida <i>Feature Branch</i>	19
Figura 6 – Formulário para consulta de NF-e	22
Figura 7 – Resultado parcial de uma consulta com <i>web crawling</i>	23
Figura 8 – Captura de tela de consulta NF-e	23
Figura 9 – Resultado da extração de dados da Figura 8	24
Figura 10 – Captura de tela de consulta NF-e com filtro de saturação	24
Figura 11 – Resultado da extração de dados da Figura 10	24
Figura 12 – Captura de tela de consulta NF-e com filtro de saturação e <i>threshold</i> .	25
Figura 13 – Resultado da extração de dados da Figura 12	25
Figura 14 – Diagrama Entidade Relacionamento	27
Figura 15 – Resposta da <i>Application Programming Interface</i> (API) após registro de usuário	28
Figura 16 – Registro de um celular via Postman	31
Figura 17 – Compartilhamento de visualização de registro com proteção de dados	32
Figura 18 – Constantes de similaridade mínima exigida para validação do registro .	34
Figura 19 – Validação de campos de formulário de cadastro de usuário	37
Figura 20 – Instruções para cadastro de dispositivo: (a) Como obter a chave da NF- e, (b) Como obter o IMEI	38
Figura 21 – Atributos validados e não validados do dispositivo	39
Figura 22 – Descrição de celular contido em uma <i>tag span</i>	41
Figura 23 – Log de extração de dados da NF-e	41
Figura 24 – Telas de <i>login</i> e cadastro de usuário: (a) <i>Login</i> , (b) Cadastro de Usuário	42
Figura 25 – Menu lateral	43
Figura 26 – Telas meus celulares e informações do dispositivo: (a) Meus Celulares, (b) Informações do Dispositivo	44

Figura 27 – Código para visualização e compartilhamento do registro	45
Figura 28 – Tela de pesquisar dispositivo: (a) Formulário para pesquisar dispositivo, (b) Resultado da pesquisa	46
Figura 29 – Tela de resultado de pesquisa de dispositivo: (a) Atributos validados e não validados, (b) Histórico de transferências	47
Figura 30 – Tela de registro de celular: (a) Seleção de marca, (b) Seleção de modelo, (c) Informações básicas	48
Figura 31 – Telas de validação de registro de celular: (a) Registro com validação pendente, (b) Autorização de acesso à NF-e, (c) Registro em análise após captura de dados	50
Figura 32 – Tela de registro inválido	51
Figura 33 – Telas de criação de transferência: (a) Usuário destinatário, (b) Confirmação de transferência	52
Figura 34 – Tela de transferências: (a) Listagem de Transferências, (b) Transferência enviada, (c) Transferência recebida	53

LISTA DE ABREVIATURAS E SIGLAS

Siglas

API	<i>Application Programming Interface</i>
CI	<i>Continuous Integration</i>
CPF	Cadastro de Pessoas Físicas
DER	Diagrama Entidade Relacionamento
HTML	<i>Hyper Text Markup Language</i>
IMEI	<i>International Mobile Equipment Identity</i>
JSON	<i>JavaScript Object Notation</i>
NF-e	Nota Fiscal Eletrônica
OCR	<i>Optical Character Recognition</i>
ORM	<i>Object Relational Mapping</i>
SGDB	Sistema de Gerenciamento de Banco de Dados

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Objetivos	12
1.1.1	Objetivo geral	12
1.1.2	Objetivos específicos	12
2	TRABALHOS RELACIONADOS	13
3	MATERIAIS E MÉTODOS	17
3.1	Materiais	17
3.1.1	PHP e Laravel	17
3.1.2	MySQL e Eloquent ORM	17
3.1.3	Docker Engine e Laravel Sail	18
3.1.4	React Native e Expo	18
3.1.5	React Native Elements	18
3.1.6	Git e GitHub	18
3.2	Métodos	19
3.2.1	Fluxo de desenvolvimento	19
3.2.2	Experimentos para captura de informações da nota fiscal eletrônica	21
<u>3.2.2.1</u>	<u>Equipamentos utilizados nos experimentos</u>	21
<u>3.2.2.2</u>	<u>Tecnologias utilizadas nos experimentos</u>	21
<u>3.2.2.3</u>	<u>Experimentos com <i>web crawling</i></u>	22
<u>3.2.2.4</u>	<u>Experimentos com OCR</u>	23
<u>3.2.2.5</u>	<u>Conclusão dos experimentos</u>	24
4	DESENVOLVIMENTO DO SISTEMA	26
4.1	Modelagem do Sistema	26
4.1.1	Análise e levantamento de requisitos	26
4.1.2	Modelagem do banco de dados	27
4.2	Desenvolvimento da API	27
4.2.1	Registro de usuário	28
4.2.2	Atualização de perfil de usuário	29
4.2.3	Registro de celular	29
4.2.4	Compartilhamento de visualização de registro de celular	30

4.2.5	Transferência de registro de celular	31
4.2.6	Validação de registro de celular através de dados extraídos da NF-e	32
4.2.7	Exclusão de registro de celular	35
4.3	Desenvolvimento da Aplicação Móvel	35
4.3.1	Cadastro e autenticação de usuário	36
4.3.2	Atualização de perfil do usuário	36
4.3.3	Cadastro de dispositivo	36
4.3.4	Visualização e compartilhamento de registro	37
4.3.5	Transferência de registro do dispositivo	39
4.3.6	Extração de dados da NF-e	40
5	RESULTADOS	42
6	CONCLUSÃO	54
6.1	Trabalhos futuros	54
	REFERÊNCIAS	55

1 INTRODUÇÃO

Em 1990, iniciaram-se as operações da rede móvel de celulares no Brasil. Neste primeiro ano, foram comercializadas mil unidades desses dispositivos. Uma das primeiras grandes alterações para tornar a telecomunicação acessível ocorreu em 1998, com a privatização do sistema Telebras e a regulamentação do pré-pago, quando empresas de telefonia iniciaram suas operações no país, criando concorrência no setor supracitado. Naquele ano, o número de aparelhos celulares já era de 7,2 milhões, chegando à marca de 15 milhões no final de 1999 (ANATEL, 1999; ANATEL, 2003, p. 24).

Ano a ano, o número de acessos a esses aparelhos pela população brasileira foi crescendo, e em 2007 esse número chegou a 120,9 milhões, tornando o Brasil o 5º país que mais consumia esse tipo de tecnologia (ANATEL, 2007, p. 60). Ainda naquele ano, a empresa Apple lançou o iPhone, um aparelho que trazia diversos recursos inovadores para a época, sendo categorizado como um *smartphone*.

Em 2023, o Brasil alcançou a marca de 249 milhões de *smartphones* em operação no país, quantidade que, se dividida pelo número da população, resulta na média de 1,2 aparelhos por habitante (MEIRELLES, 2023, p. 140). Com a popularização dos *smartphones* e da Internet, novas aplicações foram sendo desenvolvidas para esses aparelhos e adotadas por seus usuários. Assim, toda uma cadeia de atividades foi sendo englobada nesses dispositivos (ANNIE, 2021, p. 5).

Ao englobar tantos recursos, como aplicativos de *delivery*, locomoção urbana e bancos, os usuários tornaram-se muito dependentes desses aparelhos, fazendo-os parecer uma extensão do corpo humano. Sendo assim, algo indispensável no dia a dia e que está sempre em posse do proprietário. Conseqüentemente, tornaram-se um alvo para a criminalidade, onde furtos, roubos e até mesmo latrocínios ocorrem para a subtração dos mesmos. O problema ganha mais evidência considerando-se o alto valor agregado que alguns aparelhos possuem (BASTOS, 2022).

Segundo o Fórum Brasileiro de Segurança Pública, em 2021 foram roubados ou furtados 847.313 celulares no Brasil, aproximadamente 2.321 aparelhos por dia (FBSP, 2022, p. 108). A maioria dos objetos furtados ou roubados tem como destino a venda informal para terceiros que, em geral, estão alheios à atividade criminosa que deu origem aos dispositivos em questão. Esses itens oriundos de crimes podem passar despercebidos no grande volume de vendas pela Internet. Segundo pesquisa realizada pela Confederação Nacional de Dirigentes Lojistas, que ouviu 958 pessoas, aproximadamente 33,9% dos entrevistados compraram *smartphones* usados pela Internet em 2021 (CNDL, 2021, p. 11). Outro possível destino é a comercialização em estabelecimentos físicos, onde também são desmontados para que suas peças sejam revendidas.

Utilizando-se de meios informais de negociação, como por aplicativos de mensagem ou até mesmo telefone, os únicos critérios de julgamento de compra acabam sendo a palavra do

vendedor e as informações que ele fornece no anúncio. Tentando validar essas informações, alguns compradores podem pesquisar mais a fundo o perfil do vendedor na plataforma de venda, pedindo também a nota fiscal do produto. Processo sem garantias e que nem todos os compradores têm facilidade para realizar.

Buscando oferecer melhorias nesse processo e mais segurança ao comprador, um sistema que informa a procedência, características e a atual propriedade de um celular pode se mostrar útil no momento da decisão de compra. Evitando que compradores se coloquem em situações vulneráveis, onde, por exemplo, anúncios falsos são criados para atrair os interessados a um local onde seus bens são subtraídos e transferências bancárias forçadas são realizadas aos criminosos.

Alguns sistemas, como por exemplo: Bike Registrada (REGISTRADA, 2022), Olho no Carro (CARRO, 2022), Celular Legal (ANATEL, 2022) e Celular Seguro BR (MJSP, 2024), já estão disponíveis online para ajudar as pessoas no momento da compra de um produto. Entretanto, não foi possível encontrar até o momento, um sistema para rastreabilidade de celulares que ofereça as funcionalidades desejadas. Sendo assim, com a criação de um sistema onde usuários possam cadastrar seus dispositivos e compartilhar o registro por meio de aplicativos de mensagem, redes sociais ou plataformas de vendas, espera-se prover aos usuários uma fonte confiável de informações que possam ser utilizadas na negociação de celulares.

1.1 Objetivos

1.1.1 Objetivo geral

Desenvolver um sistema para a rastreabilidade de celulares.

1.1.2 Objetivos específicos

- Permitir o cadastro de usuários;
- Permitir que os usuários registrem celulares que possuam nota fiscal eletrônica vinculada ao seu CPF;
- Permitir que os usuários gerenciem registros de dispositivos vinculados à sua conta;
- Permitir que os usuários visualizem e compartilhem dados de celulares já cadastrados;
- Permitir que os usuários visualizem o histórico dos proprietários de um celular;
- Permitir que os usuários proponham e aceitem transferências de propriedades de celulares.

2 TRABALHOS RELACIONADOS

Esta seção apresenta algumas plataformas utilizadas para gerenciamento ou consultas de registros de bens de consumo, sendo elas: Bike Registrada (REGISTRADA, 2022), Olho no Carro (CARRO, 2022), Celular Legal (ANATEL, 2022) e Celular Seguro BR (MJSP, 2024). A pesquisa para identificar trabalhos relacionados foi realizada nas plataformas Google e Google Acadêmico, utilizando os seguintes termos: “sistema para rastreabilidade de celulares”, “sistema para rastreabilidade de celulares utilizando nota fiscal”, “rastreabilidade de celulares”, “histórico de proprietários de um celular”, “sistema para identificar o proprietário de um *smartphone*” e “sistema para identificar o proprietário de um celular”. Entretanto, até o momento, não foi possível encontrar trabalhos relacionados que abordem a rastreabilidade de celulares por meio do compartilhamento de informações entre usuários e do histórico de transferências de propriedade.

O Bike Registrada (REGISTRADA, 2022) é um sistema que traz uma grande quantidade de recursos para usuários que realizam o cadastro de sua bicicleta no site. A Figura 1 apresenta alguns recursos destacados pela plataforma, sendo possível perceber ênfase em segurança, combate ao comércio de bicicletas roubadas e no histórico de transferências de propriedade.

Figura 1 – Tela inicial Bike Registrada



Fonte: (REGISTRADA, 2022).

O sistema Olho no Carro (CARRO, 2022), auxilia os usuários da plataforma a obter informações detalhadas sobre o veículo pesquisado. A aplicação trata-se de um serviço pago, onde o usuário deve se cadastrar no sistema, escolher um dos planos de cobertura e realizar o pagamento. A Figura 2 apresenta um exemplo de consulta no sistema, sendo possível identificar que o veículo pesquisado possui histórico de passagem por leilões e restrições na sua documentação, sendo elas de nível nacional e estadual, o que pode acarretar na impossibilidade de transferência do veículo para outra pessoa.

Figura 2 – Modelo de consulta Olho no Carro

The screenshot displays two panels for vehicle restrictions and a table for auction history.

Restrições nacionais

Renajud	CONSTA BLOQUEIO JUDICIAL - RENAJUD
ENTENDA	

Restrições estaduais

Renajud	CONSTA BLOQUEIO JUDICIAL - RENAJUD
Administrativa	SINISTRADO/RECUPERADO
ENTENDA	

São as restrições que constam na base de dados do Detran, como a Restrição Administrativa, a Restrição Benefício Tributário e a Restrição Roubo/Furto. Algumas dessas restrições dificultam a transferência do veículo para o seu nome!
Para saber mais [clique aqui](#) e leia um texto completo sobre este assunto no nosso blog.

Lelão

Data Lelão	ID Lelão	Lote	Placa	Chassi	Marca	Modelo	Condições	Comtente
22/08/2013	-	0	AAA0000	00AA00011AAA000AA1	JEEP	RENEGADE LGT T AT	-	NC
22/08/2013	-	0	AAA0000	00AA00011AAA000AA1	JEEP	RENEGADE LGT T AT	-	NC
22/08/2013	-	0	AAA0000	00AA00011AAA000AA1	JEEP	RENEGADE LGT T AT	-	NC
22/08/2013	-	0	AAA0000	00AA00011AAA000AA1	JEEP	RENEGADE LGT T AT	-	NC

ENTENDA

Fonte: (CARRO, 2022).

Celular Legal (ANATEL, 2022) é um projeto que através da informação, visa combater a compra de celulares roubados ou adulterados. Utilizando o IMEI do dispositivo, o usuário pode consultar na plataforma se o aparelho possui registro de roubo ou furto. A Figura 3 ilustra o resultado de uma consulta para um IMEI pertencente a um aparelho roubado. Este aparelho não possui restrições no sistema, devido ao proprietário não ter realizado o boletim de ocorrência e comunicado a operadora sobre o ocorrido, impedindo que a informação fosse registrada no sistema.

Celular Seguro BR (MJSP, 2024) é um aplicativo que foi desenvolvido com o intuito de combater o roubo e furto de aparelhos celulares. Para utilizá-lo, o usuário deve cadastrar seus aparelhos no sistema, e em caso de roubo, furto ou perda, é possível emitir um alerta que bloqueia o celular, tornando-o inutilizável. O aplicativo também notifica instituições bancárias parceiras, que podem bloquear temporariamente as contas do usuário, aumentando a segurança e dificultando o uso indevido do aparelho por terceiros. A Figura 4 apresenta o registro de um aparelho na plataforma, exibindo sua situação atual e as opções disponíveis para gerenciar o registro, como editar informações, excluir o aparelho ou emitir um alerta de bloqueio.

Das quatro aplicações mencionadas, três fornecem aos seus usuários informações sobre o objeto pesquisado, possibilitando que suas decisões de compra sejam embasadas nesses

Figura 3 – Consulta de IMEI Celular Legal

Consulta Celular Legal

Última atualização
28/10/2022 21:46:54

Confira a situação do seu aparelho celular

IMEI

Não sou um robô

IMEI	Resultado	Data Consulta	Responsável
355084694712493	Até o momento o IMEI informado não possui restrições de uso	28/10/2022 21:47:15	

Fonte: (ANATEL, 2022).

Figura 4 – Tela de gerenciamento de aparelhos na versão web do Celular Seguro BR.

gov.br | Governo Federal
Ola, BRUNO JOSE DOS SANTOS WOGT

Ministério da Justiça e Segurança Pública
Celular Seguro

Registrar Telefone

Cadastrar Telefone

Pesquisar

Telefone	Operadora	Marca	Protocolo	Pessoas de Confiança	Situação	Ações
(00) 90000-0000	Tim	Apple			Sem Restrição	

Fonte: (MJSP, 2024).

dados. O Bike Registrada (REGISTRADA, 2022), embora permita a consulta de bicicletas cadastradas, não realiza a validação desses dados, levantando dúvidas sobre a confiabilidade das informações contidas no registro. Isso exige que os usuários realizem etapas complementares, como a verificação da nota fiscal da bicicleta. Outro ponto é que o usuário pode vincular uma imagem da nota fiscal ao registro, mas sua visualização é pública, permitindo que dados sensíveis do proprietário fiquem expostos para todos.

O Olho no Carro (CARRO, 2022) centraliza informações sobre veículos que estão dispersas em outros sistemas e entrega aos usuários conforme demanda, estas informações são importantes pois podem refletir diretamente na negociação, transferência de registro e valor do veículo.

No caso do Celular Legal (ANATEL, 2022), a ferramenta é útil para identificar se o IMEI consultado pertence a um aparelho roubado ou que possui alguma restrição, mas não é possível

identificar o atual proprietário e as características do aparelho. Também é possível que a base de dados não possua o registro de roubo de um celular caso o proprietário não tenha informado o ocorrido e realizado o bloqueio do dispositivo.

Já o Celular Seguro BR (MJSP, 2024) oferece recursos para evitar que o aparelho seja utilizado após o seu roubo, mas não permite o compartilhamento de informações sobre o registro do dispositivo com terceiros. Além disso, não é possível realizar a transferência deste registro, sendo necessário que o antigo proprietário o exclua para que o novo proprietário possa vincular o aparelho à sua conta no aplicativo.

Com isso, justifica-se a implementação de uma nova plataforma que utilize os pontos fortes identificados nos trabalhos relacionados, como por exemplo: consulta de um registro, histórico de transferências, características do produto, e realize a validação dos dados informados pelo usuário, através da nota fiscal eletrônica, pois este documento possui diversos dados que podem ser usados como parâmetros para validação do dispositivo.

3 MATERIAIS E MÉTODOS

3.1 Materiais

O desenvolvimento web possui um catálogo diversificado de tecnologias que podem ser empregadas mutuamente dependendo das necessidades do projeto. Para a definição das tecnologias a serem utilizadas, foram observados os seguintes pontos: a complexidade dos recursos a serem desenvolvidos, prazos de entregas, comportamento e características esperadas da aplicação, tempo de mercado, e por fim a familiarização do autor deste projeto com as mesmas.

3.1.1 PHP e Laravel

Para o *backend* do sistema, foi desenvolvida uma API utilizando a linguagem de programação PHP¹ e o *framework* Laravel², devido ao conjunto de ferramentas que este oferece para a construção de APIs robustas e de fácil configuração a partir de seus kits iniciais. Esse conjunto inclui recursos como migrações de banco de dados, *Object Relational Mapping* (ORM), sistema de roteamento intuitivo, *middleware* para tratamento de requisições, validação de dados integrada, mecanismos de autenticação e autorização (LARAVEL, 2024d).

Essa escolha também se justifica pela ampla comunidade de desenvolvedores e pela vasta documentação disponível, sendo assim, encontrar suporte para problemas relacionados a essas tecnologias é facilitado pela grande quantidade de fóruns e materiais disponíveis online.

3.1.2 MySQL e Eloquent ORM

Para o gerenciamento de dados, foi definido o uso do Sistema de Gerenciamento de Banco de Dados (SGDB) MySQL³, por se tratar de uma solução *open source*, robusta e amplamente utilizada. O MySQL garante a integridade dos dados através do relacionamento entre tabelas, evitando exclusões acidentais que poderiam gerar registros inconsistentes. Além disso, o MySQL é uma das opções primárias suportadas pelo *framework* Laravel, o que facilita a integração e o desenvolvimento da aplicação (VIEIRA, 2023; LARAVEL, 2024a)

Visando aumentar a segurança e produtividade ao trabalhar com a base de dados e extrair o máximo do ecossistema Laravel, optou-se pelo uso do Eloquent⁴, um ORM que mapeia os objetos do sistema através de modelos e os associa as tabelas no banco de dados, deste modo abstraindo a complexidade para construção e utilização de recursos baseados nestes relacionamentos (LARAVEL, 2024b).

¹ <https://www.php.net/>

² <https://laravel.com/>

³ <https://www.mysql.com/>

⁴ <https://laravel.com/docs/11.x/eloquent>

3.1.3 Docker Engine e Laravel Sail

Buscando consistência entre os ambientes de desenvolvimento e, futuramente, o de produção, optou-se pelo uso do Docker Engine⁵, uma plataforma que permite empacotar aplicações e suas dependências em contêineres, garantindo assim a portabilidade e a replicabilidade da configuração. Juntamente a ele, foi empregado o uso do Laravel Sail⁶, uma interface de linha de comando que simplifica a gestão do ambiente de desenvolvimento baseado em contêineres do Docker. Deste modo, foi possível configurar de forma mais rápida e eficiente as tecnologias do *backend* deste projeto, como: PHP, Laravel e MySQL (IBM, 2024; LARAVEL, 2024c).

3.1.4 React Native e Expo

O *frontend* do sistema, implementado como aplicação móvel, foi desenvolvido utilizando React Native⁷, um *framework* que combina a biblioteca React com APIs nativas do Android e iOS. O desenvolvimento foi direcionado para Android, aproveitando as capacidades específicas desse sistema. Isso permitiu criar uma base de código em TypeScript⁸, um superconjunto do JavaScript⁹, que com ajustes, pode ser interpretada em ambos os sistemas. A utilização do React Native foi facilitada pelo Expo¹⁰, um *framework* que encapsula ferramentas e recursos nativos, além de abstrair a complexidade da configuração do ambiente de desenvolvimento e do processo de *build* da aplicação (CUNHA, 2022; EXPO, 2022).

3.1.5 React Native Elements

Para otimizar e padronizar a construção da interface do aplicativo, optou-se pelo uso do React Native Elements¹¹ como biblioteca de componentes visuais. A escolha se justifica pela vasta gama de elementos pré-construídos e personalizáveis oferecidos pela biblioteca, o que agiliza o desenvolvimento e garante uma interface consistente.

3.1.6 Git e GitHub

Para o versionamento de código foi utilizado o Git¹², um sistema que possibilita a criação de contextos intitulados de ramificações, usados para isolar o desenvolvimento de recursos,

⁵ <https://docs.docker.com/get-started/overview/>

⁶ <https://laravel.com/docs/11.x/sail>

⁷ <https://reactnative.dev/>

⁸ <https://www.typescriptlang.org/>

⁹ <https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>

¹⁰ <https://expo.dev/>

¹¹ <https://reactnativeelements.com/>

¹² <https://git-scm.com/>

testes ou manutenções no código. Deste modo, ao submeter estas ramificações para o repositório remoto do projeto na plataforma GitHub¹³, torna-se possível adicionar novos recursos e correções ao sistema.

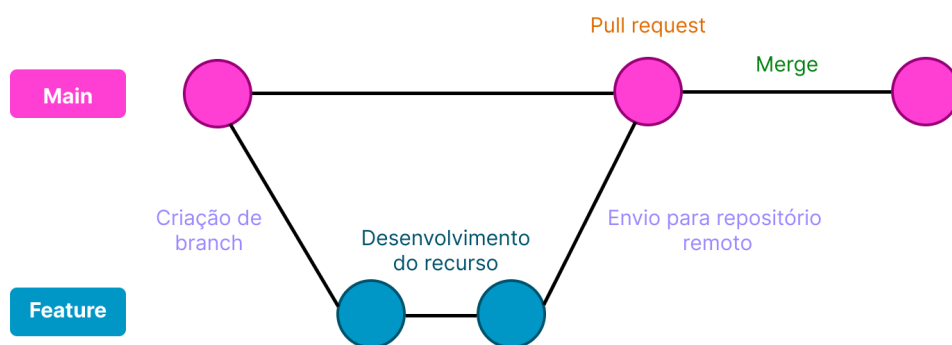
Visando automatizar processos de *Continuous Integration* (CI), optou-se pelo uso do GitHub Action¹⁴, uma ferramenta para criar tarefas que são executadas através de um gatilho, que pode ser, por exemplo, a criação de um *Pull Request* no repositório remoto. Deste modo quando este evento ocorre, alguns processos são iniciados, como por exemplo: testes unitários para garantir que o código está funcionando adequadamente e sua integração com a ramificação principal não coloca em risco a integridade do sistema (GIT, 2022; GITHUB, 2022).

3.2 Métodos

3.2.1 Fluxo de desenvolvimento

Tratando-se do fluxo de trabalho no Git, foi escolhida a metodologia *Feature Branch Workflow* (BITBUCKET, 2022), que define processos para isolar o desenvolvimento de novos recursos da ramificação principal. Na Figura 5 é possível visualizar o ciclo de vida de uma *Feature Branch*, que consiste nas seguintes etapas: (1) criação de *branch* baseada na ramificação principal e desenvolvimento do recurso, (2) envio da *branch* local para o repositório remoto e criação de *Pull Request* para análise do orientador, por fim (3) mesclagem do novo recurso com a ramificação principal.

Figura 5 – Ciclo de vida *Feature Branch*



Fonte: Autoria própria (2024).

Para o desenvolvimento do projeto o fluxo de trabalho escolhido baseia-se na metodologia ágil, onde os requisitos do sistema são levantados, analisados e divididos em tarefas. Processo que se repete ao decorrer do desenvolvimento. As atividades foram categorizadas

¹³ <https://github.com/>

¹⁴ <https://docs.github.com/pt/actions>

conforme sua prioridade de desenvolvimento, sendo elas: Essencial (recursos vitais para o funcionamento do sistema), Importante (recursos que complementam o sistema), Desejável (recursos extras).

No gerenciamento das atividades de desenvolvimento dos requisitos, utilizamos o GitHub Projects¹⁵, com quadros Kanban separados por repositório *backend* e *frontend*. Os quadros são divididos em quatro raias, sendo elas: *TODO* (a fazer), *DOING* (em andamento), *REVIEW* (aguardando análise do orientador) e *DONE* (finalizado). As atividades de cada repositório são inicialmente adicionadas à raia *TODO*, sinalizando as tarefas que precisam ser realizadas. Essa organização permite acompanhar o progresso do projeto de forma visual e eficiente.

O fluxo de desenvolvimento de cada requisito segue as seguintes etapas:

1. Selecionar um requisito pertencente a raia *TODO* no quadro Kanban, que possua a maior prioridade de desenvolvimento;
2. Mover recurso para a aba *DOING*, sinalizando que o mesmo está no processo de desenvolvimento;
3. Criar uma *branch* local no ambiente de desenvolvimento para encapsular do restante da aplicação os trabalhos no novo recurso (início da etapa 1 do fluxo *Feature Branch*);
4. Analisar e identificar a necessidade de criação ou alteração de entidades no banco de dados;
5. Construir o recurso;
6. Executar testes (fim da etapa 1 do fluxo *Feature Branch*);
7. Enviar a *branch* local da atividade para o repositório remoto do projeto no GitHub (início da etapa 2 do fluxo *Feature Branch*);
8. Criar uma *Pull Request* no repositório do projeto no Github adicionando o professor orientador como revisor;
9. Executar os testes automatizados na *Pull Request* através do Github Actions;
10. Mover recurso no quadro Kanban para raia *Review*;
11. Aguardar análise do professor orientador (fim da etapa 2 do fluxo *Feature Branch*);
12. Caso necessário, realizar correções apontadas pelo orientador ou pelos testes automatizados;

¹⁵ <https://docs.github.com/pt/issues/planning-and-tracking-with-projects/learning-about-projects/about-projects>

13. Se o passo anterior for verdadeiro, então é necessário retornar o fluxo para o item 4 e mover recurso no quadro Kanban para raia *Doing*;
14. Finalizado, realizar *merge* na ramificação principal (etapa 3 do fluxo *Feature Branch*);
15. Mover recurso no quadro Kanban para raia *Done*, sinalizando que o mesmo foi finalizado.

3.2.2 Experimentos para captura de informações da nota fiscal eletrônica

A validação do registro de um dispositivo é um processo vital no sistema, o compartilhamento das informações com terceiros é habilitado somente após esta etapa. Este processo é necessário para confirmar a veracidade das informações cadastradas pelos usuários. Para obter as informações da NF-e referente à compra do dispositivo cadastrado no sistema, foram levantadas inicialmente duas alternativas: o desenvolvimento de um *web crawling* no *backend* para automatização de consultas e capturas de dados no site oficial da nota fiscal eletrônica; e a utilização do mecanismo *Optical Character Recognition* (OCR) para extração de dados textuais de imagens capturadas a partir da consulta da NF-e, realizada pelo usuário. Buscando identificar a abordagem mais adequada, foram realizados experimentos com ambas as técnicas. A seguir, são apresentados os resultados destes experimentos.

3.2.2.1 Equipamentos utilizados nos experimentos

- Computador com processador Ryzen 5 3600, 16GB de memória ram, 512GB de armazenamento interno e monitor 24" com resolução 1920x1080 pixels;
- Celular Redmi Note 11S, 6GB de memória ram, 64GB de armazenamento interno, tela de 6.4" com resolução de 2400x1080 pixels.

3.2.2.2 Tecnologias utilizadas nos experimentos

Visando agilizar a experimentação e validação das alternativas, optou-se por não utilizar a linguagem de programação PHP e o *framework* Laravel nos experimentos iniciais, focando em tecnologias que permitissem uma configuração mais rápida e com as quais o autor possuía maior experiência naquele momento. Para os experimentos de *web crawling*, foram utilizados

Node.js¹⁶, o *framework* Express¹⁷ e a biblioteca Puppeteer¹⁸. Já para os experimentos com OCR, foram utilizados Node.js, Tesseract OCR¹⁹ e Tesseract OCR for Node.js²⁰.

3.2.2.3 Experimentos com *web crawling*

Para implementação do *web crawling*, foi necessária a configuração de um servidor web, para possibilitar a interação do usuário com os recursos. O maior obstáculo encontrado para automatização de consultas foi o hCaptcha²¹, um mecanismo para impedir que robôs interajam com a página. Como podemos identificar na Figura 6, existe um *widget* que ao ser clicado exibe um quebra-cabeça para o usuário solucionar e em caso de sucesso um *token* oculto é adicionado ao formulário para liberação do acesso a consulta.

Figura 6 – Formulário para consulta de NF-e

Fonte: (BRASIL, 2022).

Buscando contornar o obstáculo do hCaptcha e viabilizar o *web crawling*, um experimento foi conduzido com o objetivo de obter um *token* válido através da resolução do quebra-cabeça. A abordagem consistiu em renderizar o hCaptcha para o usuário, que, ao resolvê-lo, gerava um *token* que era repassado para o *backend*. Com o uso da biblioteca Puppeteer, um navegador Chromium²² era iniciado para realizar as interações programadas e efetivar a consulta e captura dos dados na plataforma da NF-e, com o retorno sendo exemplificado na Figura 7. Após a realização de algumas tentativas, percebeu-se que o mecanismo do hCaptcha, ao identificar atividades suspeitas, entra em ação bloqueando esta automatização, inviabilizando sua aplicação no contexto desejado.

¹⁶ <https://nodejs.org/en>

¹⁷ <https://expressjs.com/pt-br/>

¹⁸ <https://pptr.dev/>

¹⁹ <https://github.com/tesseract-ocr/tesseract>

²⁰ <https://github.com/zapolnoch/node-tesseract-ocr>

²¹ <https://www.hcaptcha.com/>

²² <https://www.chromium.org/chromium-projects/>

Figura 7 – Resultado parcial de uma consulta com *web crawling*

```

Data de Emissão
05/09/2022 21:12:19-03:00

Data/Hora de Saída ou da Entrada
05/09/2022 22:12:21-03:00

Valor Total da Nota Fiscal
98,79
Emitente
CNPJ
09.339.936/0007-01

Nome / Razão Social
NS2.COM INTERNET S.A.

Inscrição Estadual
049150464
  
```

Fonte: Autoria própria (2024).

3.2.2.4 Experimentos com OCR

Para construção da técnica utilizando OCR, foi necessária a configuração de um ambiente de testes, para extração de dados das capturas de telas realizadas no site oficial da nota fiscal eletrônica através do aparelho celular Redmi Note 11S. A Figura 8 apresenta uma captura realizada com a orientação da tela do dispositivo na vertical, resultando após o corte em uma imagem de 1080x368 pixels. Os dados da imagem de teste foram extraídos em formato de texto e exibidos no terminal, onde alguns erros podem ser identificado em vermelho, como mostra a Figura 9.

Figura 8 – Captura de tela de consulta NF-e

Num.	Descrição	Qtd.	Unidade Comercial	Valor(R\$)
1	CASE C3 TECH P HD 2.5 PRETO - CH-200BK - TOT TRIB. 8.01	1,0000	UN	33,30
2	COOLER PARA PROCESSADOR DEEPCOOL GAMMAXX GT V2, AMDINTEL - DP-MCH4-GMX-GTV2 - TOT TRIB. 32.74	1,0000	UN	180,41

Fonte: (BRASIL, 2022).

O primeiro teste para otimizar está técnica foi aplicar um filtro para remover a saturação da imagem, tornando-a cinza, pois o fundo da mesma tem um tom alaranjado o que pode prejudicar o reconhecimento dos caracteres. A Figura 10 demonstra este processo, com o resultado da extração de dados resultando em um único erro, o qual está destacado em vermelho na Figura 11.

Figura 9 – Resultado da extração de dados da Figura 8

```

• brunow@pop-os:~/Documents/Projetos DEV/Project0crNode$ node app.js
NFe Emitente Destinatário Produtos e Serviços Totais Transporte Cobrança Informações Adicionais

Dados dos Produtos e Serviços

[ Num. Descrição ara. Unidade Comercial Valor(R$)
| 1 | CASE C3 TECH P HD 2.5 PRETO - CH-200BK - TOT TRIB. 8.01 | 1,0000 | UN | 3330

2 | COOLER PARA PROCESSADOR DEEPCOOL GAMMAXX GT V2, AMDINTEL - | 1,0000 UN 180,41
DP-MCH4-GMX-GTV2 - TOT TRIB. 32.74

```

Fonte: Autoria própria (2024).

Figura 10 – Captura de tela de consulta NF-e com filtro de saturação

Num.	Descrição	Qtd.	Unidade Comercial	Valor(R\$)
1	CASE C3 TECH P HD 2.5 PRETO - CH-200BK - TOT TRIB. 8.01	1,0000	UN	33,30
2	COOLER PARA PROCESSADOR DEEPCOOL GAMMAXX GT V2, AMDINTEL - DP-MCH4-GMX-GTV2 - TOT TRIB. 32.74	1,0000	UN	180,41

Fonte: Autoria própria (2024).

Figura 11 – Resultado da extração de dados da Figura 10

```

• brunow@pop-os:~/Documents/Projetos DEV/Project0crNode$ node app.js
NFe Emitente Destinatário Produtos e Serviços Totais Transporte Cobrança Informações Adicionais

Dados dos Produtos e Serviços

| Num. Descrição Qtd. Unidade Comercial Valor(R$)
| 1 CASE C3 TECH P HD 2.5 PRETO - CH-200BK - TOT TRIB. 8.01 | 1,0000 UN | 33,30

2 | COOLER PARA PROCESSADOR DEEPCOOL GAMMAXX GT V2, AMDINTEL - | 1,0000 UN 180,41
DP-MCH4-GMX-GTV2 - TOT TRIB. 32.74

```

Fonte: Autoria própria (2024).

O segundo teste para otimizar esta técnica foi capturar somente a descrição dos produtos e aplicar os filtros de saturação e *threshold* para deixar a imagem em preto e branco. A Figura 12 demonstra este processo, que resultou em uma imagem de 1065x514 pixels que, ao ter seus dados extraídos, não gerou erros, como mostra a Figura 13.

3.2.2.5 Conclusão dos experimentos

Com o desenvolvimento dos experimentos, foi possível testar ambas das alternativas levantadas como possíveis soluções para obter dados da NF-e, identificando pontos positivos e negativos em cada técnica. O *web crawling* demonstrou ser uma excelente opção quando se trata

Figura 12 – Captura de tela de consulta NF-e com filtro de saturação e *threshold*

Emitente	Destinatário	Produtos e Serviços	Totais	Transporte	Cobrança
Dados dos Produtos e Servi					
m. Descrição					
1		CASE C3 TECH P HD 2.5 PRETO - CH-200BK - TOT TRIB. 8.01			
2		COOLER PARA PROCESSADOR DEEPCOOL GAMMAXX GT V2, AMDINTEL - DP-MCH4-GMX-GTV2 - TOT TRIB. 32.74			

Fonte: Autoria própria (2024).

Figura 13 – Resultado da extração de dados da Figura 12

```

● brunow@pop-os:~/Documents/Projetos DEV/Project0crNode$ node app.js
Emitente Destinatário Produtos e Serviços Totais Transporte Cobrança

Dados dos Produtos e Servi

m. Descrição

1 CASE C3 TECH P HD 2.5 PRETO - CH-200BK - TOT TRIB. 8.01

2 COOLER PARA PROCESSADOR DEEPCOOL GAMMAXX GT V2, AMDINTEL -
DP-MCHA4-GMX-GTV2 - TOT TRIB. 32.74

```

Fonte: Autoria própria (2024).

da quantidade de interações necessárias entre o usuário e o sistema para alcançar o objetivo, bem como a qualidade dos resultados da consulta, mas, quando implementado no *backend*, é suscetível a bloqueios, como observado nos experimentos. O OCR demonstrou ter resultados promissores, apresentando alguns erros que podem ser minimizados ou solucionados completamente utilizando filtros para melhorias das imagens, mas necessita de uma maior interação entre o usuário e o sistema, além do direcionamento ao usuário sobre as informações que ele deve capturar. Também é necessário atentar-se sobre potenciais riscos na veracidade das imagens submetidas para validação e as diferentes resoluções de telas dos dispositivos.

Concluindo, ambas as técnicas alcançaram os objetivos propostos de capturar informações da NF-e. Contudo, o *web crawling* no *backend*, devido ao seu bloqueio, demonstrou não ser viável, abrindo espaço para explorar sua implementação no *frontend*, como alternativa para contornar o bloqueio do hCaptcha e a complexidade de utilizar o mecanismo OCR. A implementação do *web crawling* no *frontend* é apresentada na seção 4.3.6 (Extração de dados da NF-e).

4 DESENVOLVIMENTO DO SISTEMA

Este capítulo apresenta as etapas do desenvolvimento do sistema, detalhando as atividades e os desafios encontrados durante a modelagem e implementação da API e da aplicação móvel.

4.1 Modelagem do Sistema

4.1.1 Análise e levantamento de requisitos

A análise e o levantamento de requisitos foi realizada com base nos resultados do desenvolvimento da pesquisa e experimentos conduzidos ao longo deste trabalho. Através desta abordagem foi possível identificar os principais requisitos funcionais que o sistema deveria atender, os quais foram posteriormente organizados e priorizados em quadros Kanban nos repositórios remotos do projeto no GitHub, facilitando o acompanhamento e a gestão do desenvolvimento.

No Quadro 1 estão descritos os requisitos funcionais necessários para o sistema.

Quadro 1 – Requisitos Funcionais

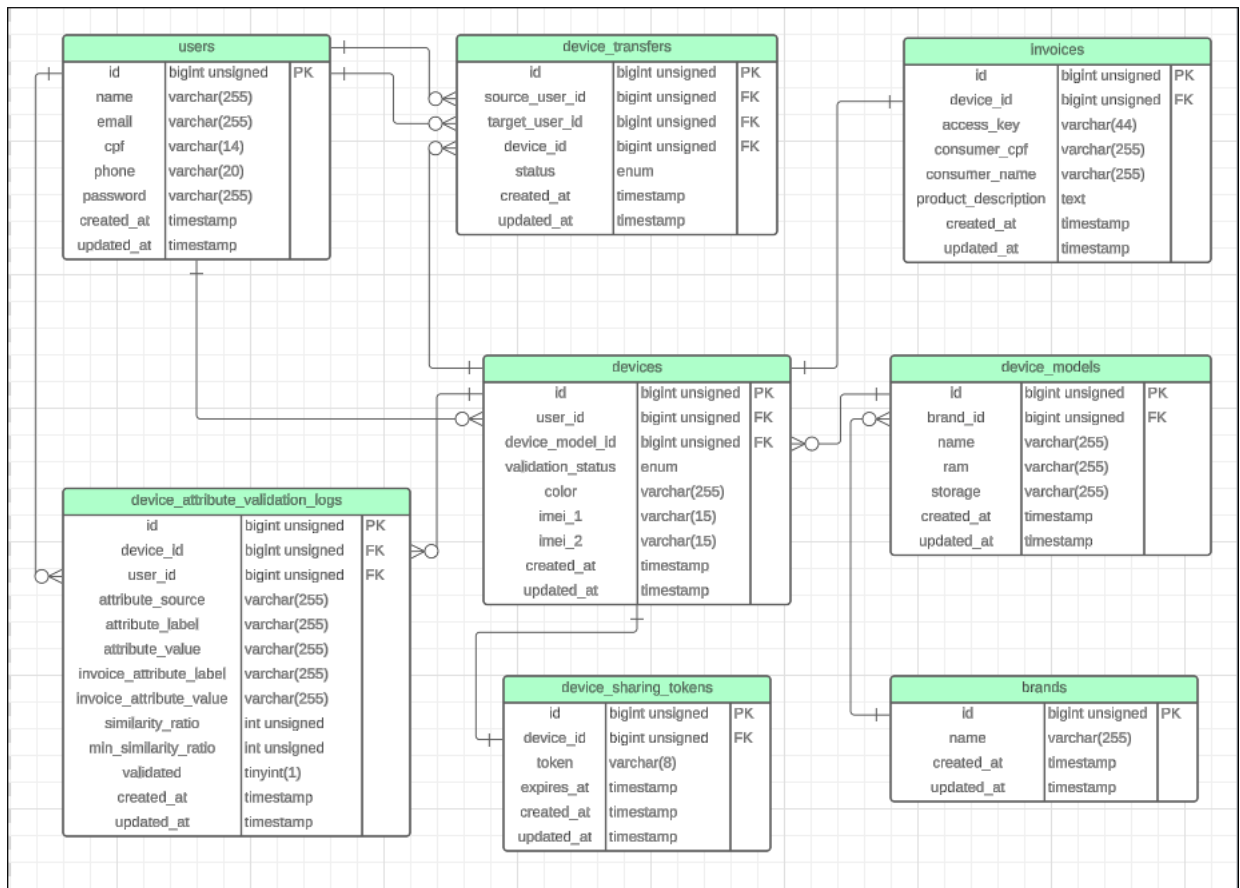
Requisito	Descrição	Prioridade
RF001	Cadastro de usuários	Essencial
RF002	Autenticação utilizando email e senha	Essencial
RF003	<i>Logout</i> de usuário autenticado	Essencial
RF004	Recuperação de acesso a conta	Essencial
RF005	Cadastro de dispositivos	Essencial
RF006	Validação do registro do dispositivo	Essencial
RF007	Compartilhamento do registro do dispositivo entre usuários	Essencial
RF008	Transferência de registro de dispositivo entre usuários	Essencial
RF009	Exclusão de registros inválidos	Importante
RF010	Atualização cadastral do usuário no sistema	Importante
RF011	Geração de <i>pin</i> para confirmação de transferência de registro	Desejável
RF012	Cadastro de usuário utilizando conta Google	Desejável
RF013	Bloqueio de compartilhamento do registro pelo usuário	Desejável

Fonte: Autoria própria (2024).

4.1.2 Modelagem do banco de dados

Baseando-se na análise e levantamento de requisitos, foi possível desenvolver o Diagrama Entidade Relacionamento (DER), como mostra a Figura 14. O DER ilustra a estrutura do banco de dados, definindo as entidades, seus atributos e os relacionamentos entre elas.

Figura 14 – Diagrama Entidade Relacionamento



Fonte: Autoria própria (2024).

4.2 Desenvolvimento da API

Para o desenvolvimento da API do sistema, foi utilizado o Docker para orquestrar os serviços da aplicação. Primeiramente, configurou-se o ambiente Docker e, em seguida, criou-se um projeto Laravel utilizando o Sail, conforme sua documentação oficial¹. Visando agilizar o desenvolvimento e focar nas funcionalidades específicas do sistema, optou-se pelo uso do Breeze², um *scaffold* do Laravel que já provê recursos de autenticação e autorização.

A adoção do Breeze possibilitou que os esforços fossem concentrados nas funcionalidades chave do sistema, eliminando a necessidade de desenvolver toda a estrutura de auten-

¹ <https://laravel.com/docs/10.x/installation#sail-on-linux>

² <https://laravel.com/docs/10.x/starter-kits#laravel-breeze>

ticação desde o início. Após a configuração inicial do projeto, o código foi versionado em um repositório remoto, com o nome de “ta-safe-api”³, criado na plataforma GitHub.

4.2.1 Registro de usuário

O Laravel Breeze gera automaticamente o recurso de registro de usuário, mas este mostrou-se insuficiente para atender às necessidades do sistema. Diante disso, foram realizadas modificações para adequá-lo aos requisitos específicos.

Inicialmente, a lógica de registro estava concentrada em um *controller*. Visando melhorar a organização e a divisão de responsabilidades, essa lógica foi transferida para uma *action* dedicada ao registro de novos usuários. Para isso, foi criado um *form request* com os campos adicionais necessários (CPF e telefone) e suas respectivas validações. No *model* do usuário, foi implementado um método que instancia e executa a *action*, responsável pela lógica de criação de usuários e validação das regras de negócio.

A fim de garantir o correto funcionamento do recurso após as alterações, foram implementados testes unitários e de *controller*. Além disso, para padronizar e controlar a resposta da API, foi utilizado o Laravel *api resources*, que permite formatar a resposta em *JavaScript Object Notation* (JSON), conforme ilustrado na Figura 15.

Figura 15 – Resposta da API após registro de usuário

```
{
  "message": {
    "type": "success",
    "text": "Usuário registrado com sucesso."
  },
  "user": {
    "id": 5,
    "name": "Bruno Wogt",
    "email": "bruno@gmail.com",
    "cpf": "271.813.748-77",
    "phone": "(42) 90000-0001",
    "created_at": "2024-06-06T00:35:29.000000Z",
    "updated_at": "2024-06-06T00:35:29.000000Z",
    "token": "2|2CCnp1rMblIUZRhEuDSuiih3pD6vEgqwytFc5abv"
  }
}
```

Fonte: Autoria própria (2024).

Com a implementação dessas modificações, o recurso de registro de usuários passou a atender plenamente às necessidades do projeto.

³ <https://github.com/bruw/ta-safe-api>

4.2.2 Atualização de perfil de usuário

Visando possibilitar a atualização de dados básicos do usuário, foi implementado um recurso para edição do perfil. Através deste, o usuário pode modificar seu nome, email e telefone. O CPF, contudo, permanece inalterável, pois é utilizado para validar registros de celulares. A implementação deste recurso não apresentou desafios significativos.

4.2.3 Registro de celular

Para viabilizar o registro de celulares no sistema, inicialmente foi necessário definir quais modelos de aparelhos seriam suportados. A diversidade de marcas e modelos no mercado brasileiro tornou essa definição crucial, pois gerenciar cada dispositivo individualmente seria inviável para o escopo do projeto, uma vez que o tratamento do conjunto completo de dispositivos oferece desafios, como uma grande quantidade de aparelhos com nomenclatura similar ou ainda variações de um mesmo modelo, neste momento inicial, acredita-se que isso poderia impactar negativamente na assertividade e no cruzamento de informações com a NF-e.

A seleção dos modelos foi baseada em dados de mercado sobre os celulares seminovos mais vendidos em 2022, conforme veiculado pela plataforma Tecnoblog (2023). A lista inclui modelos populares como o iPhone 7, iPhone 11, iPhone XR, entre outros. No entanto, embora o iPhone 7 seja o líder de vendas de celulares usados, ele foi lançado em 2016, tornando menos provável que os usuários ainda tenham a NF-e de compra do aparelho. Portanto, buscando o equilíbrio entre popularidade e probabilidade do aparelho ainda possuir a NF-e, optou-se por uma seleção restrita de aparelhos, abrangendo diferentes marcas e modelos, conforme apresentado no Quadro 2.

Após essa definição, foram criados *seeders* para registrar as marcas e modelos na base de dados. Vale ressaltar que cada modelo possui submodelos que se diferenciam pela configuração de *hardware*, como memória RAM e armazenamento interno. Para todos os modelos listados no Quadro 2, foram registrados os principais submodelos comercializados no Brasil.

Em seguida, foi implementada a estrutura para registro de celulares no sistema. Para isso, foram gerados *models* para representar o dispositivo, a marca e a nota fiscal. Para a criação das tabelas correspondentes no banco de dados, foram criadas *migrations*, e um *controller* específico foi desenvolvido para gerenciar as requisições de registro destes aparelhos. A lógica de registro foi encapsulada em uma *action*, e um *form request* valida a entrada de dados submetidas pelo usuário. Seguindo o padrão de desenvolvimento adotado, foram implementados testes unitários e de *controller* para garantir a qualidade do código. A Figura 16 ilustra um exemplo de requisição e resposta deste recurso utilizando a ferramenta Postman⁴.

⁴ <https://www.postman.com/>

Quadro 2 – Modelos de Celulares Suportados

Marca	Modelo
Apple	iPhone 11
Apple	iPhone 12
Apple	iPhone 13
Motorola	Edge 20
Motorola	Edge 30
Samsung	Galaxy A31
Samsung	Galaxy S20
Samsung	Galaxy S21
Samsung	Galaxy S22
Xiaomi	Poco X3 Pro
Xiaomi	Poco X4 Pro
Xiaomi	Poco X5 Pro

Fonte: Autoria própria (2024).

Após o registro, o aparelho fica com o status de validação pendente, sendo necessário que o usuário realize a validação para liberar o acesso a todas as funcionalidades do sistema, pois registros neste estado não podem ser compartilhados ou transferidos para outros usuários.

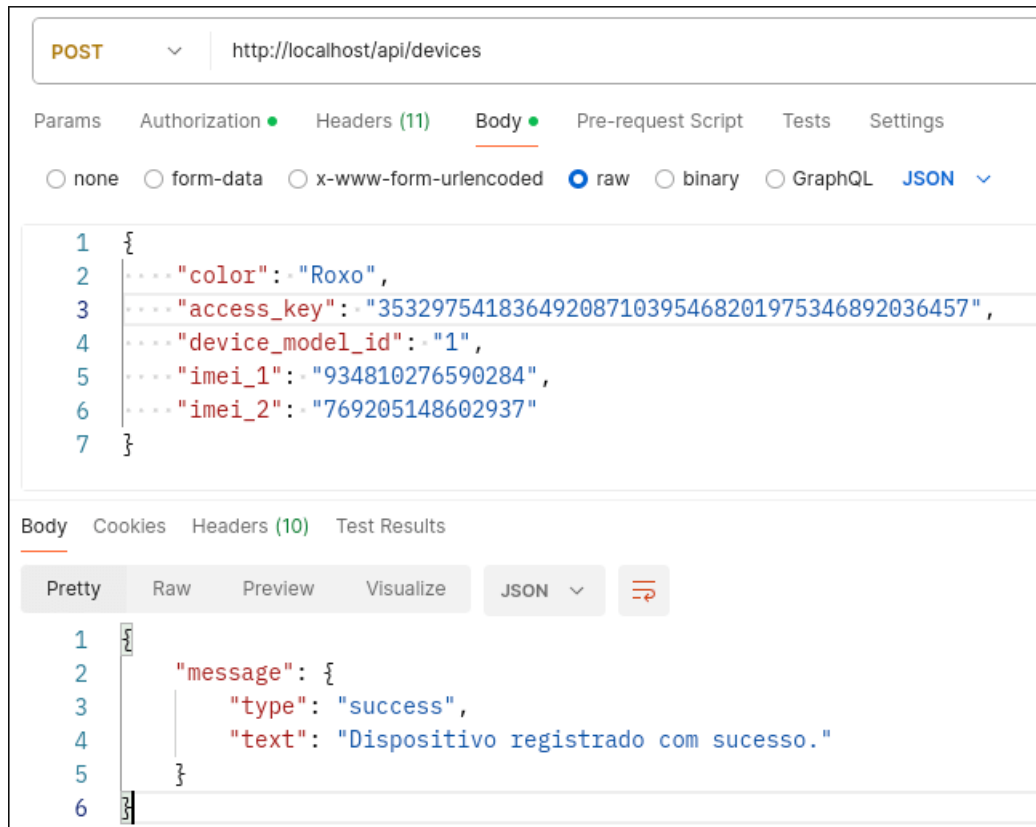
4.2.4 Compartilhamento de visualização de registro de celular

O compartilhamento do registro é uma funcionalidade que permite ao usuário proprietário de um dispositivo validado no sistema compartilhar a visualização das informações do aparelho com terceiros. Para isso, foi implementada uma *action* que gera *tokens* aleatórios para visualização temporária.

Cada *token* possui 8 dígitos, sendo únicos no sistema. Além disso, possui um prazo de validade de 24 horas e não é possível ter mais de um código ativo para o mesmo registro. Após sua expiração, o *token* se torna inválido, sendo necessária a geração de um novo, que automaticamente substitui o anterior. A limitação da validade do *token* garante que um usuário tenha acesso a um registro compartilhado consigo apenas durante o período necessário para a negociação do aparelho, evitando assim o acesso por tempo indeterminado a esse registro.

Para garantir a privacidade dos dados sensíveis durante a visualização, como o CPF e telefone dos usuários, bem como o IMEI do aparelho, foi implementada uma máscara que oculta parte desses detalhes. Assim, os campos sensíveis são visualizados parcialmente, conforme demonstrado na Figura 17.

Figura 16 – Registro de um celular via Postman



Fonte: Autoria própria (2024).

Com um código válido em mãos, o usuário proprietário pode compartilhá-lo em plataformas de vendas ou aplicativos de mensagens e redes sociais, permitindo que outros usuários, que também estejam cadastrados no sistema, acessem informações relevantes sobre o aparelho, como marca, modelo, cor e histórico de proprietários.

4.2.5 Transferência de registro de celular

A transferência de registro de um celular é um recurso que permite atualizar o proprietário do aparelho e gerar um novo elo na sua rastreabilidade. Para iniciar uma transferência, o usuário proprietário deve criar uma proposta, selecionando o dispositivo a ser transferido e o usuário destinatário. Para isso, foi criada uma *migration* que gera uma tabela específica na base de dados para armazenar as transferências, registrando os usuários envolvidos, o dispositivo e o status da transferência (pendente, aceita, cancelada, rejeitada). Além disso, foi criado um *model* para representar uma transferência e um *controller* que recebe a requisição, valida seus atributos por meio de um *form request* e repassa as informações para uma *action* que efetiva a criação da proposta após a validação das regras de negócio. Caso alguma regra não seja atendida, uma exceção é gerada com uma mensagem específica que é retornada ao usuário. A proposta de transferência é gerada com o status “pendente”, e ambos os usuários envolvidos podem modificá-la enquanto ela estiver nesse status.

Figura 17 – Compartilhamento de visualização de registro com proteção de dados

```

{
  "id": 1,
  "color": "Preto",
  "imei_1": "548*****285",
  "imei_2": "302*****592",
  "validation_status": "validated",
  "created_at": "2024-06-08T20:28:54.000000Z",
  "updated_at": "2024-06-08T20:28:54.000000Z",
  "user": {
    "id": 1,
    "name": "Monkey D. Luffy",
    "cpf": "27*****04",
    "phone": "(42)*****001",
    "created_at": "2024-06-08T20:28:54.000000Z"
  },
  "device_model": {
    "id": 7,
    "name": "Galaxy S22",
    "ram": "8 GB",
    "storage": "256 GB",
    "brand": {
      "id": 3,
      "name": "Samsung"
    }
  }
},
  "validation_attributes": {

```

Fonte: Autoria própria (2024).

Após a criação da proposta, o usuário destinatário tem a opção de aceitá-la ou rejeitá-la. Se a proposta for aceita, o sistema efetiva a transferência do registro para o novo proprietário, atualizando o histórico do aparelho e modificando o status para “aceito”. Caso seja rejeitada ou o atual proprietário decida cancelar a proposta de transferência antes da aceitação, o registro do celular permanece inalterado e nenhuma mudança é feita no histórico do aparelho, apenas no status da transferência, que é modificado para “rejeitado” ou “cancelado”, respectivamente. Essa flexibilidade permite que os usuários tenham controle sobre o processo e possam desistir da operação caso necessário.

4.2.6 Validação de registro de celular através de dados extraídos da NF-e

A validação do registro de um celular no sistema é uma etapa crucial para garantir a autenticidade das informações cadastradas pelo usuário. No *backend*, esse processo é realizado utilizando os dados extraídos da NF-e pelo aplicativo móvel, conforme detalhado na subseção 4.3.6 (Extração de dados da NF-e).

A validação consiste em verificar se o nome e CPF do consumidor na nota fiscal correspondem aos dados do usuário. Além disso, a nota fiscal deve conter a descrição do dispositivo, confirmando que se refere ao aparelho em validação.

Devido a impossibilidade de comparar diretamente as informações do usuário e os atributos do dispositivo cadastrado com os dados extraídos da nota fiscal, foi necessário aplicar a técnica de correspondência difusa de *strings*, através da biblioteca FuzzyWuzzy⁵, que permite comparar *strings* e calcular a similaridade entre elas, mesmo que haja pequenas variações, como erros de digitação ou abreviações. Essa biblioteca possui diversos métodos que podem ser empregados em diferentes cenários, mas para as necessidades deste recurso apenas dois métodos foram utilizados, sendo eles: *ratio* e *tokenSetRatio*. O método *ratio* calcula a similaridade entre duas *strings* utilizando comparações simples de manipulação para transformar a menor na maior, considerando a ordem de ambas. Já o método *tokenSetRatio* divide as *strings* em *tokens*, para então calcular a similaridade, desconsiderando a ordem das palavras. Ambos dos métodos geram uma pontuação entre 0 e 100 pontos. (BATHULA, 2023; PYKES, 2023).

Com exceção do CPF, que utilizou a comparação direta de *strings*, os demais atributos (nome do usuário, marca, modelo, cor, IMEI, memória RAM e armazenamento do dispositivo) foram validados utilizando a correspondência difusa de *strings*. Para validar se o nome do usuário corresponde ao nome do consumidor da NF-e, optou-se pelo uso do método *ratio*, por ser uma comparação entre duas *strings* em que o tamanho e a ordem dos caracteres são importantes. Já para os demais atributos, que dizem respeito às características do dispositivo, foi necessário o uso do método *setTokenRatio*, pois era preciso verificar se cada atributo em validação estava presente na descrição do produto, independentemente de sua posição na *string*.

Para cada atributo do registro, foi definida uma constante que sinaliza o índice mínimo de similaridade necessária para validá-lo. Essas constantes foram definidas em uma classe própria, conforme mostrado na Figura 18. Os valores foram definidos com base na importância de cada atributo no registro. Por exemplo, o nome do consumidor na nota fiscal pode estar abreviado, então ao compará-lo com o nome por extenso do usuário, o resultado de similaridade pode apresentar diferenças, mas ainda assim ser aceitável. Quanto maior o índice mínimo de similaridade exigida, mais restritiva é a validação.

Devido à possibilidade de a nota fiscal conter mais de um produto, definir qual deles representa um celular tornou-se um grande desafio. Para solucionar este problema e extrair da lista de produtos a descrição que mais se aproxima do dispositivo em validação, foi desenvolvida uma *action* que encapsula a lógica para analisar linha por linha dos produtos contidos na NF-e, utilizando o método *setTokenRatio* da biblioteca FuzzyWuzzy e aplicando pesos aos valores calculados de similaridade para cada atributo verificado.

Conforme ilustrado no Quadro 3, o modelo do dispositivo tem o maior peso no cálculo de similaridade, pois sua presença na descrição de um produto na NF-e aumenta significativamente a probabilidade de a linha que está sendo analisada conter a descrição de um celular. Atribuindo pesos maiores à memória RAM e ao armazenamento interno (comparados à marca e cor), foi possível evitar falsos positivos em descrições ambíguas, como “Celular iPhone 11 16GB 128GB branco” e “Capa para iPhone 11 branca”. Visando aprimorar este processo, foi definida

⁵ <https://github.com/wyndow/fuzzywuzzy>

uma pontuação mínima de 795 pontos para a similaridade total da linha, já multiplicada pelos pesos dos atributos. Esse valor garante que, se os atributos de maior peso estiverem presentes na descrição da NF-e, a pontuação será facilmente alcançada, identificando e extraindo corretamente a linha que contém a descrição do produto. Caso nenhuma linha atinja a pontuação mínima, é retornada uma *string* vazia.

Figura 18 – Constantes de similaridade mínima exigida para validação do registro

```
<?php
namespace App\Constants;

class DeviceAttributeValidationRatio
{
    /**
     * Name similarity must be equal to or greater than 75%.
     */
    public const MIN_NAME_SIMILARITY = 75;

    /**
     * Brand name similarity must be equal to or greater than 75%.
     */
    public const MIN_BRAND_SIMILARITY = 75;

    /**
     * Device model name similarity must be equal to or greater than 85%.
     */
    public const MIN_MODEL_NAME_SIMILARITY = 85;

    /**
     * Device color similarity must be equal to or greater than 70%.
     */
    public const MIN_COLOR_SIMILARITY = 70;

    /**
     * Device ram size similarity must be equal to or greater than 70%.
     */
    public const MIN_RAM_SIMILARITY = 70;

    /**
     * Device storage size similarity must be equal to or greater than 70%.
     */
    public const MIN_STORAGE_SIMILARITY = 70;

    /**
     * Device IMEIS similarity must be equal to or greater than 90%.
     */
    public const MIN_IMEI_SIMILARITY = 90;
}
```

Fonte: Autoria própria (2024).

Após a identificação da linha que contém a descrição do produto na NF-e, ela é repassada para as respectivas *actions*, responsáveis por validar cada atributo do registro do celular. Cada *action* compara a *string* base, que representa uma característica do celular, com a linha extraída da nota fiscal. Em seguida, registra o resultado da validação no banco de dados.

Devido à inconsistência na forma como um dispositivo é descrito na nota fiscal, é provável que haja ausência de alguns atributos na descrição. Portanto, foi necessário definir quais teriam sua validação obrigatória para que o registro fosse considerado válido. Assim, definiu-se que o nome e CPF do proprietário, marca e modelo do dispositivo deveriam ser validados com sucesso. Além disso, pelo menos uma das características técnicas, como a memória RAM ou

Quadro 3 – Pesos dos atributos na identificação do produto na NF-e

Atributo do dispositivo	Peso
Modelo	6,0
Memória RAM	3,0
Armazenamento interno	3,0
Cor	1
Marca	1

Fonte: Autoria própria (2024).

o armazenamento interno, também deve ser validada. Com esses atributos validados, é possível afirmar que aquele usuário possui o dispositivo descrito, embora alguns parâmetros não tenham sido validados com sucesso, sendo necessário sinalizar de forma clara estas inconsistências aos usuários que acessarem este registro.

4.2.7 Exclusão de registro de celular

Após a reprovação de um registro na validação, é necessário permitir ao usuário a exclusão desse registro. Essa funcionalidade é importante para que, caso o usuário tenha cometido erros no cadastro e/ou na validação, ele possa reutilizar a mesma NF-e, já que o sistema não permite que uma mesma chave de acesso de nota fiscal seja utilizada em mais de um registro.

Para implementar a exclusão do registro, foi criado um *controller* que recebe e processa a requisição, encaminhando as informações para uma *action* responsável por excluir o dispositivo, desde que as regras de negócio sejam atendidas. Somente registros com status de validação “rejeitado” podem ser excluídos. Caso um registro com status diferente tente ser excluído, ou qualquer outra regra de negócio não seja atendida, uma mensagem explicando o erro é retornada ao usuário e a ação é interrompida. O desenvolvimento deste recurso, assim como o de atualização de perfil de usuário, não apresentou desafios significativos.

4.3 Desenvolvimento da Aplicação Móvel

Para atender à crescente demanda por acessibilidade e mobilidade, o *frontend* do sistema, implementado como aplicação móvel, foi desenvolvido utilizando React Native com o *framework* Expo. Essa abordagem permite que os usuários interajam com o sistema de forma conveniente e intuitiva em seus dispositivos móveis.

O processo de desenvolvimento iniciou-se com a criação e configuração de um projeto Expo localmente, que posteriormente foi versionado em um repositório remoto, com o nome “ta-safe-mobile”⁶, na plataforma GitHub.

⁶ <https://github.com/bruw/ta-safe-mobile>

4.3.1 Cadastro e autenticação de usuário

O primeiro recurso desenvolvido no *frontend* foi o de autenticação de usuário. A atividade consistiu em desenvolver um componente para captura das credenciais de *login* do usuário e submeter estas informações para a API, que verifica os dados recebidos e retorna um *token* de acesso em caso de sucesso. Esse *token* é armazenado localmente no dispositivo do usuário e utilizado em todas as requisições subsequentes à API, garantindo a segurança e a autorização do usuário ao acesso dos recursos.

Inicialmente, construiu-se um formulário utilizando variáveis de estado para gerenciar os dados de entrada do usuário. No entanto, buscando simplificar e otimizar esse processo, optou-se pelo uso da biblioteca React Hook Form⁷. Esta biblioteca oferece recursos avançados para gerenciamento de formulários, validação de dados e manipulação de erros, tornando o desenvolvimento mais eficiente e organizado.

O mesmo se aplica para o recurso de registro de usuário, onde foi criado um formulário com os campos necessários para o cadastro de novos usuários, sendo estes: nome, email, CPF, celular, senha e confirmação de senha. Para formatação dos campos CPF e celular, foram criadas máscaras que facilitam o seu correto preenchimento, melhorando a experiência do usuário.

Todas as validações de dados são feitas no *backend*. Caso ocorra um erro, a API retorna uma mensagem de validação específica para o *frontend*, que então exibe essa mensagem no campo correspondente do formulário. Isso permite que o usuário receba um *feedback* imediato e direcionado sobre quaisquer problemas com os dados informados, conforme ilustrado na Figura 19.

4.3.2 Atualização de perfil do usuário

Para atualização do perfil do usuário foi desenvolvido um recurso para possibilitar que o mesmo visualize e edite seus dados cadastrais, sendo eles: nome, email e celular. Já o seu CPF é visível mas não editável, devido às regras de negócio da aplicação. Este recurso não apresentou desafios significativos durante a sua implementação.

4.3.3 Cadastro de dispositivo

Para o desenvolvimento deste recurso foram identificados desafios de interface, dada a quantidade de informações necessárias no registro do aparelho. Para simplificar esse processo e torná-lo mais intuitivo para o usuário, optou-se pela criação de um formulário de múltiplas etapas, que divide o processo de cadastro em partes menores e mais gerenciáveis.

⁷ <https://react-hook-form.com/>

Figura 19 – Validação de campos de formulário de cadastro de usuário

9:49

Alguns erros foram encontrados, por favor verifique-os:

Nome

O campo nome é obrigatório.

Email

O campo email é obrigatório.

CPF

O campo cpf é obrigatório.

Celular

O campo celular é obrigatório.

Senha

O campo senha é obrigatório.

Confirmação de Senha

Fonte: Autoria própria (2024).

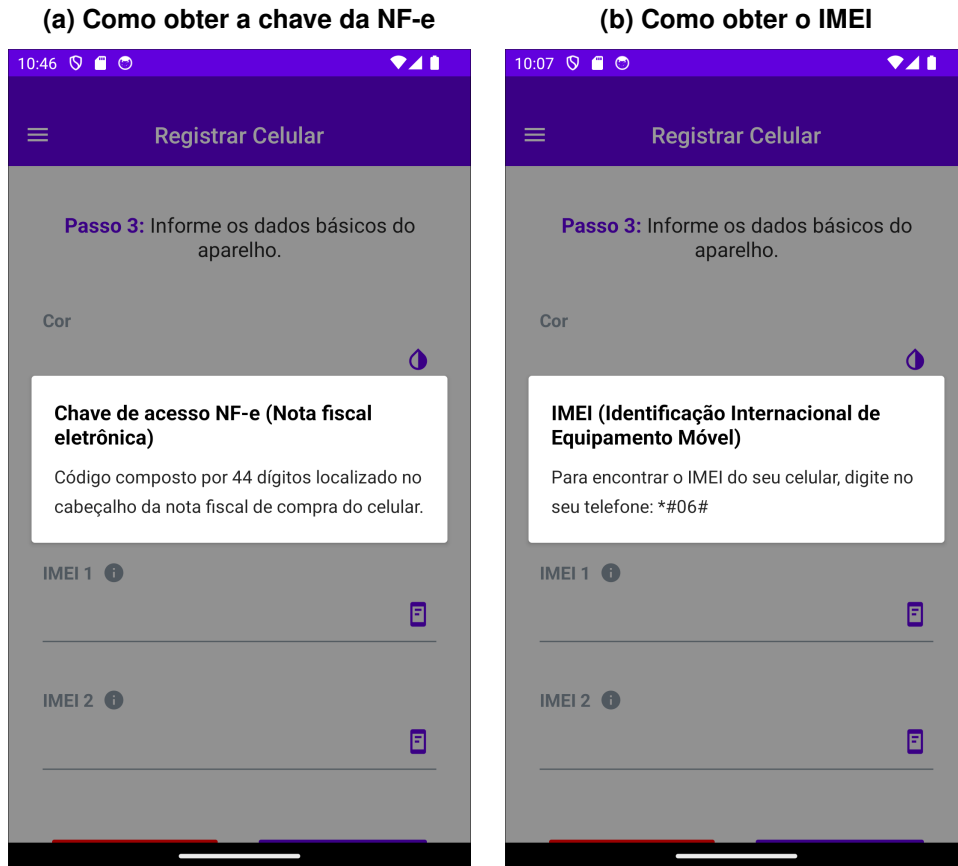
A primeira etapa do formulário inicia-se com a seleção da marca do dispositivo. Para isso, foi implementado um componente de seleção que carrega todas as marcas registradas no sistema, conforme a lista de aparelhos suportados que foi apresentada na subseção 4.2.3 (Registro de celular). Uma vez selecionada a marca, a aplicação exibe dinamicamente os modelos relacionados a ela. Após a seleção do modelo, o usuário avança para a etapa final, onde deve informar a cor do dispositivo, a chave de acesso da NF-e e os IMEIs 1 e 2 do aparelho. Para auxiliar o usuário no preenchimento correto dos campos “chave de acesso da NF-e”, “IMEI 1” e “IMEI 2”, foi construído um componente para visualização de instruções que detalham o que deve ser inserido em cada campo e como obter essas informações. A Figura 20 ilustra o uso deste componente.

Com todas as informações preenchidas, o usuário pode submeter os dados para a API, que realizará a validação e, em caso de sucesso, registrará o dispositivo, vinculando-o à conta do usuário. Em caso de erro na validação, a API retorna mensagens específicas para cada campo, permitindo que o usuário identifique e corrija os erros sinalizados.

4.3.4 Visualização e compartilhamento de registro

Após o cadastro e validação do registro do dispositivo, o usuário proprietário tem a possibilidade de compartilhar a visualização de informações do aparelho com outros usuários do

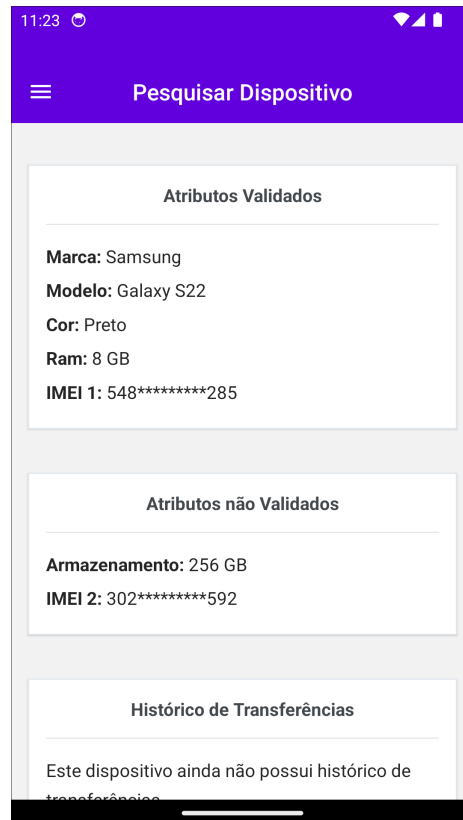
Figura 20 – Instruções para cadastro de dispositivo: (a) Como obter a chave da NF-e, (b) Como obter o IMEI



Fonte: Autoria própria (2024).

sistema. Para isso, foi necessário implementar um recurso que possibilitou a interação do aplicativo móvel com a API para geração de *tokens* de visualização temporária do registro.

Para esta visualização, foi construída uma tela específica que divide as informações do registro em *cards*, possibilitando o foco em cada aspecto do mesmo. Conforme discutido na subseção 4.2.6 (Validação de registro de celular através de dados extraídos da NF-e), o dispositivo em validação não necessita que todos os seus atributos sejam validos, sendo necessário que apenas os mais importantes estejam presentes. Deste modo, foi de extrema importância separar e enfatizar para o usuário quais atributos foram validados e quais não foram. A divisão dessas informações em áreas diferentes auxilia o usuário a entender o registro do dispositivo e, diante de uma negociação com o proprietário do aparelho, discutir sobre a veracidade das informações compartilhadas consigo. A Figura 21 ilustra esta divisão na visualização dos atributos do aparelho, onde também é possível observar a ofuscação (máscara) de dados sensíveis da NF-e.

Figura 21 – Atributos validados e não validados do dispositivo

Fonte: Autoria própria (2024).

4.3.5 Transferência de registro do dispositivo

A transferência de registro do dispositivo foi um recurso construído em algumas etapas. Na primeira etapa, foi desenvolvido um componente para o usuário proprietário iniciar uma transferência, informando o e-mail do usuário destinatário. Nesse momento, o aplicativo móvel realiza uma requisição para a API a fim de obter os dados do usuário destinatário. Caso este seja válido, as informações do dispositivo e do usuário destinatário são exibidas ao proprietário, que após confirmação, gera uma proposta de transferência.

A segunda etapa do desenvolvimento desse recurso consistiu em criar uma tela específica para visualizar todas as transferências criadas ou recebidas pelo usuário. Nela, o usuário proprietário tem a possibilidade de cancelar propostas criadas por ele que ainda não foram aceitas. Já o usuário destinatário pode rejeitar as propostas de transferências recebidas ou aceitá-las, o que o tornaria o novo proprietário do aparelho.

Um cuidado tomado no desenvolvimento desse recurso foi garantir que todas as ações referentes às transferências fossem confirmadas antes de ser efetivadas e que, caso exista um código de compartilhamento do dispositivo, ele fique disponível nos detalhes da proposta para que o usuário destinatário visualize as informações do aparelho. Caso não exista um código, o usuário destinatário é instruído a solicitar ao proprietário um *token* para visualização dos detalhes do registro.

4.3.6 Extração de dados da NF-e

A extração de dados da nota fiscal eletrônica referente à compra do aparelho celular é um recurso essencial no sistema, pois somente através deste processo é possível obter os dados para que o *backend* consiga realizar o processo de validação do registro de um celular.

Conforme apresentado na subseção 3.2.2 (Experimentos para captura de informações da nota fiscal eletrônica), foram realizados testes com *web crawling* no *backend* e o uso de algoritmos de OCR para extração de dados da NF-e, mas ambas das abordagens demonstraram limitações. Visando contorná-las e obter estes dados tão importantes para este trabalho, decidiu-se pela opção de realizar o *web crawling* no *frontend*, o que possibilita a captura das informações de uma maneira rápida e segura em relação às outras abordagens.

Para realizar este processo no aplicativo móvel, inicialmente foi necessário estudar a estrutura da NF-e e entender como o recurso de extração de dados poderia ser construído utilizando as tecnologias do *frontend*. A solução desenvolvida foi utilizar o componente `WebView`⁸ do React Native para renderizar o portal da nota fiscal eletrônica para o usuário, que ao resolver um quebra-cabeça do mecanismo hCaptcha concede o acesso as informações da NF-e. Este processo não visa burlar o sistema da nota fiscal eletrônica, apenas auxiliar o usuário que concedeu o acesso à sua NF-e a obter os dados digitais da mesma.

Visando automatizar a interação e captura destes dados, foram construídos, com JavaScript, algoritmos que utilizam seletores *Hyper Text Markup Language* (HTML) para checar os elementos da página e capturar as informações úteis. Com esta abordagem foi possível extrair os dados estritamente necessários para validação do registro, sendo eles o nome e CPF do consumidor, bem como a lista de produtos referentes a NF-e.

Um dos grandes desafios relacionados à lista de produtos é que uma nota fiscal pode conter um ou vários itens, sendo necessário extrair todas as descrições. Conforme abordado na subseção 4.2.6 (Validação de registro de celular através de dados extraídos da NF-e), o *backend* utiliza uma *action* para encontrar a linha que possui a maior probabilidade de ser a descrição do produto em validação. A primeira abordagem no *frontend* foi extrair somente o conteúdo textual dos produtos, o que demonstrou não ser uma boa ideia, pois deste modo, não tínhamos delimitações para sinalizar onde uma descrição começava ou terminava. Mas, a partir da análise desta abordagem, chegou-se à solução para resolver este problema, que foi manter as marcações HTML de cada descrição, pois em volta delas temos o uso de *tags span*, conforme ilustrado na Figura 22. Deste modo, mantendo as *tags span* é possível identificar o início e o fim de cada item na descrição da lista de produtos. A Figura 23 representa o log de extração de dados de uma nota fiscal (nome e CPF do consumidor foram omitidos devido a serem dados sensíveis).

Após a extração dos dados da NF-e, o aplicativo móvel envia estas informações para API, que atualiza o registro do dispositivo com os dados extraídos da nota fiscal eletrônica.

⁸ <https://github.com/react-native-webview/react-native-webview/tree/master>

Figura 22 – Descrição de celular contido em uma tag span

```
<tr>
  <td class="fixo-prod-serv-numero">... </td>
  <td class="fixo-prod-serv-descricao">
    <span>Celular Samsung Galaxy A31 A315 Dual 128gb 4gb Ram Azul</span> == $0
  </td>
  <td class="fixo-prod-serv-qtd">... </td>
  <td class="fixo-prod-serv-uc">... </td>
  <td class="fixo-prod-serv-vb">... </td>
</tr>
```

Fonte: Autoria própria (2024).

Figura 23 – Log de extração de dados da NF-e

```
LOG
cpf:
name:
products: <span>Celular Samsung Galaxy A31 A315 Dual 128gb 4gb Ram Azul</span>
```

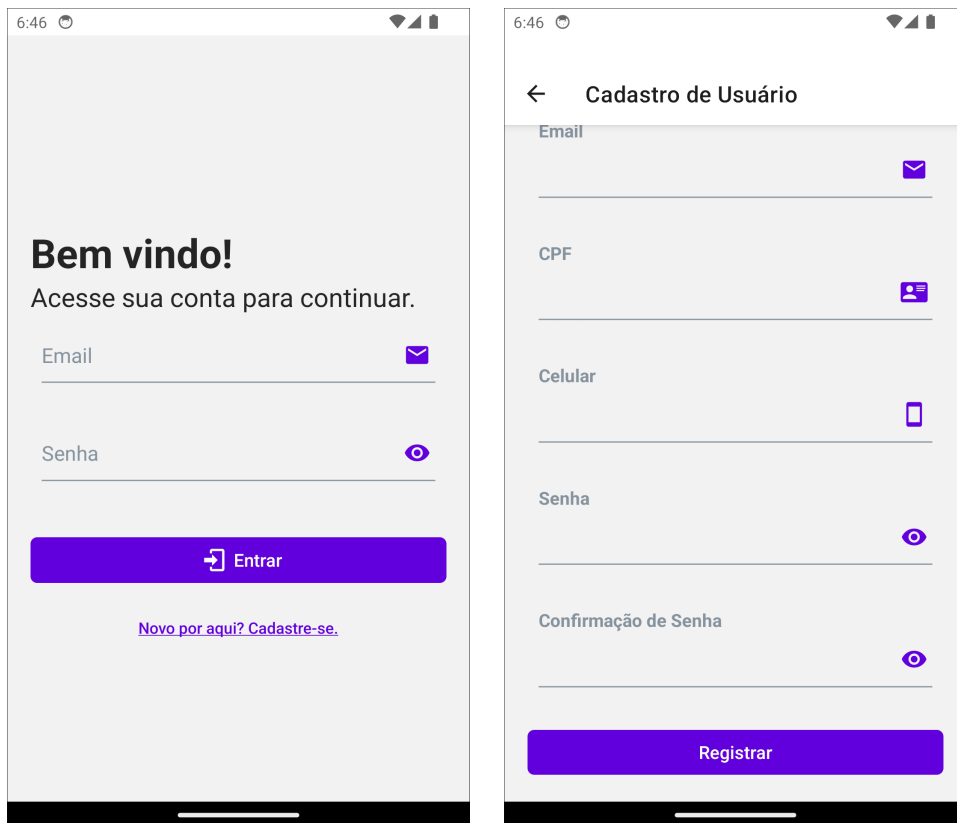
Fonte: Autoria própria (2024).

5 RESULTADOS

Nesta seção, serão apresentados os resultados obtidos com o desenvolvimento do sistema. Nela, serão exibidas capturas de tela do aplicativo móvel, demonstrando as principais funcionalidades e a interface do usuário.

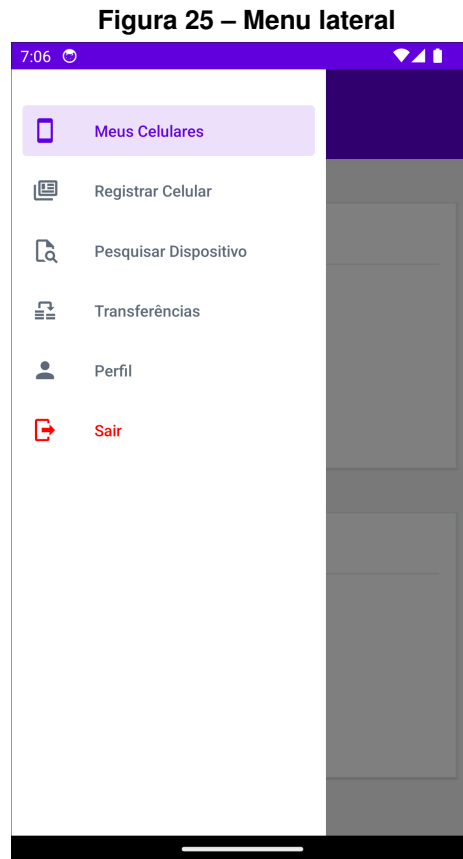
A tela de *Login* possibilita que os usuários cadastrados no sistema tenham acesso a suas contas utilizando as suas credenciais. No caso de usuários não registrados, é possível realizar o cadastro ao selecionar a opção “Novo por aqui? Cadastre-se”. Após isso, o usuário é direcionado para tela de *Cadastro de Usuário*. Ambas as telas são apresentadas na Figura 24.

Figura 24 – Telas de *login* e cadastro de usuário: (a) *Login*, (b) Cadastro de Usuário



Fonte: Autoria própria (2024).

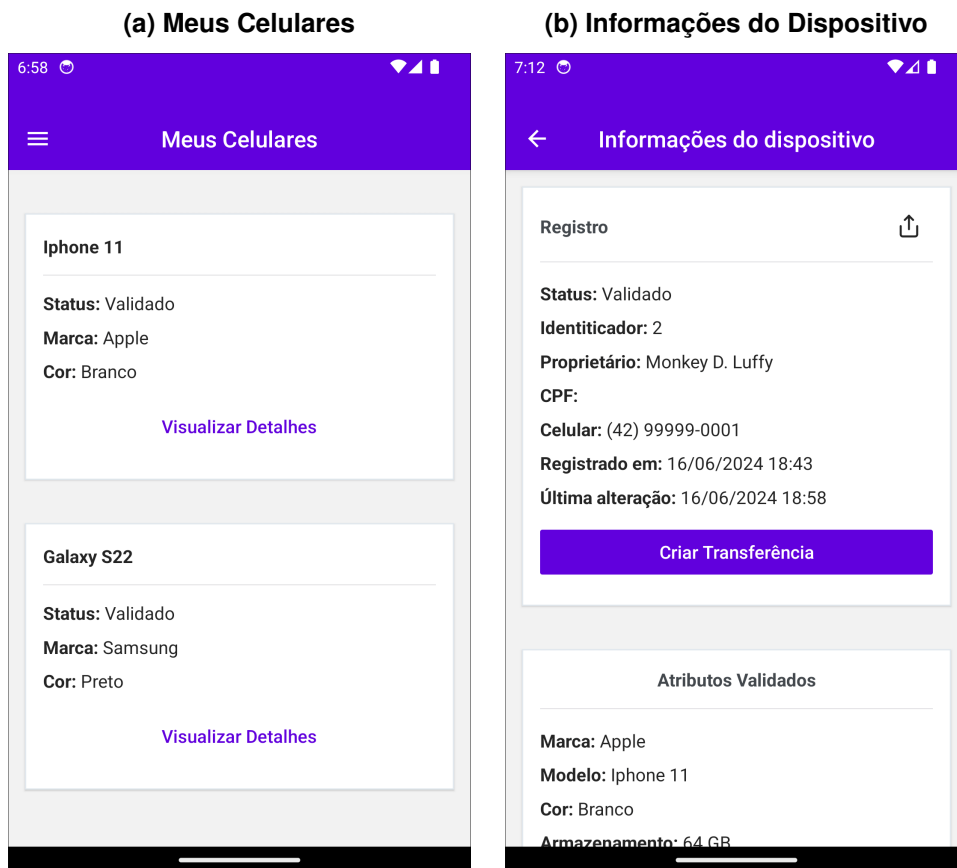
Após a sua autenticação, o usuário tem acesso a uma variedade de recursos do sistema, que podem ser acessados através do menu lateral do aplicativo, este exemplificado na Figura 25.



Fonte: Autoria própria (2024).

A primeira tela autenticada que o usuário tem acesso é a de *Meus Celulares*, que lista todos os dispositivos vinculados à sua conta. Ao selecionar a opção “Visualizar Detalhes” em um registro, o usuário é direcionado para a tela de *Informações do Dispositivo*. A Figura 26 ilustra ambas as telas.

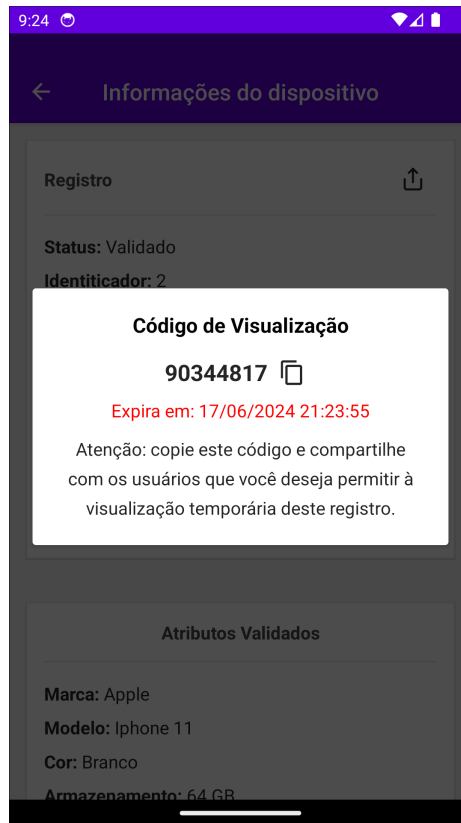
Figura 26 – Telas meus celulares e informações do dispositivo: (a) Meus Celulares, (b) Informações do Dispositivo



Fonte: Autoria própria (2024).

Na tela de *Informações do Dispositivo*, o usuário tem acesso a diversas funcionalidades relacionadas ao registro do aparelho, incluindo a opção de gerar um código de visualização e compartilhamento do registro. Este recurso está disponível somente para registros que já passaram pelo processo de validação. A Figura 27 ilustra a obtenção do código de visualização do registro.

Figura 27 – Código para visualização e compartilhamento do registro

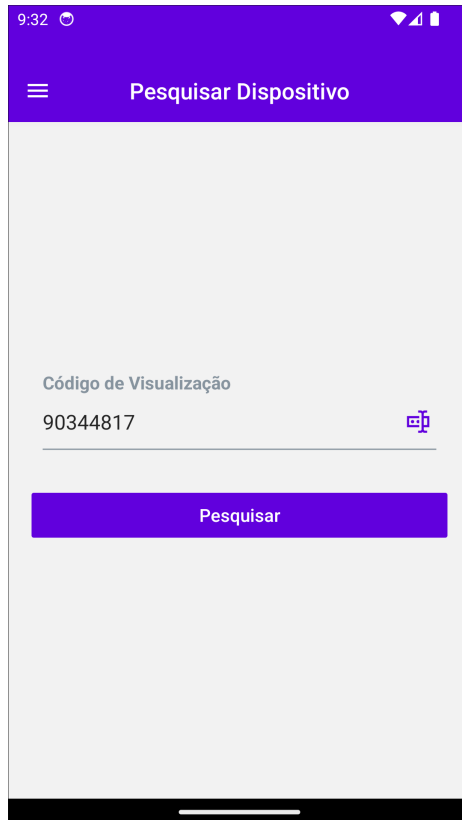


Fonte: Autoria própria (2024).

Através deste código, os usuários podem pesquisar e visualizar informações sobre um registro. Para proteger dados sensíveis dos usuários e dos aparelhos, algumas informações são parcialmente ocultadas utilizando máscaras nos resultados. A Figura 28 ilustra a tela de *Pesquisar Dispositivo*.

Figura 28 – Tela de pesquisar dispositivo: (a) Formulário para pesquisar dispositivo, (b) Resultado da pesquisa

(a) Formulário para pesquisa dispositivo



9:32

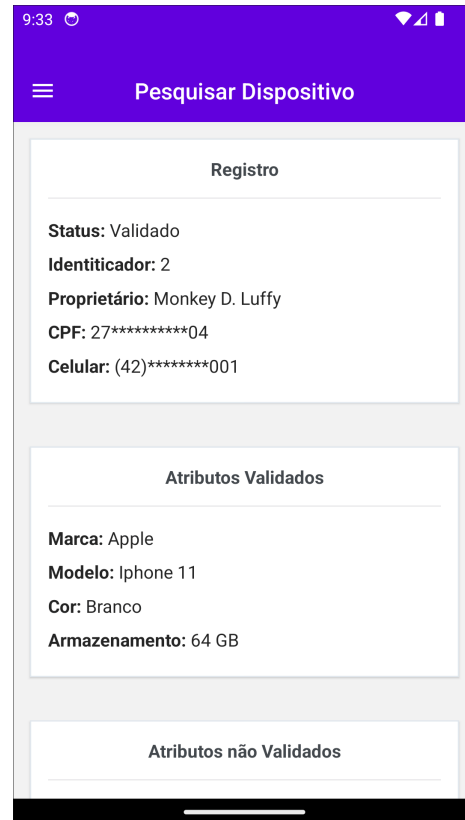
Pesquisar Dispositivo

Código de Visualização

90344817

Pesquisar

(b) Resultado da pesquisa



9:33

Pesquisar Dispositivo

Registro

Status: Validado

Identificador: 2

Proprietário: Monkey D. Luffy

CPF: 27*****04

Celular: (42)*****001

Atributos Validados

Marca: Apple

Modelo: Iphone 11

Cor: Branco

Armazenamento: 64 GB

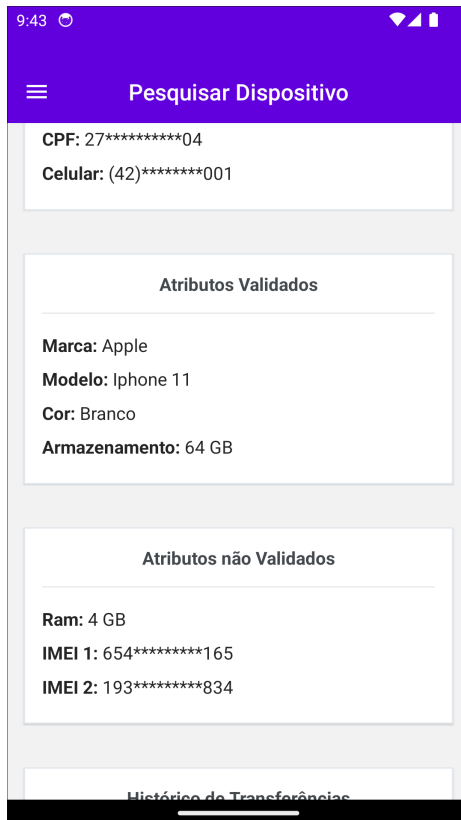
Atributos não Validados

Fonte: Autoria própria (2024).

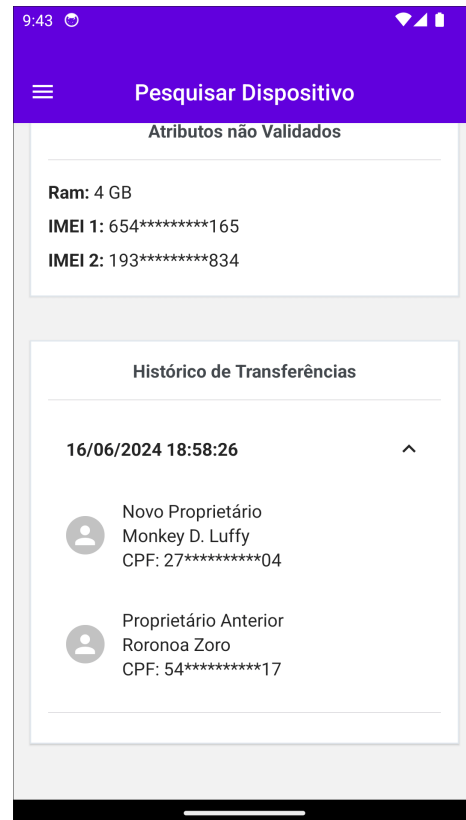
Ao visualizar um registro, o usuário obtém acesso a informações relevantes sobre o proprietário e o dispositivo, abrangendo atributos validados, não validados e o histórico de transferências, como exemplificado na Figura 29.

Figura 29 – Tela de resultado de pesquisa de dispositivo: (a) Atributos validados e não validados, (b) Histórico de transferências

(a) Atributos validados e não validados



(b) Histórico de transferências

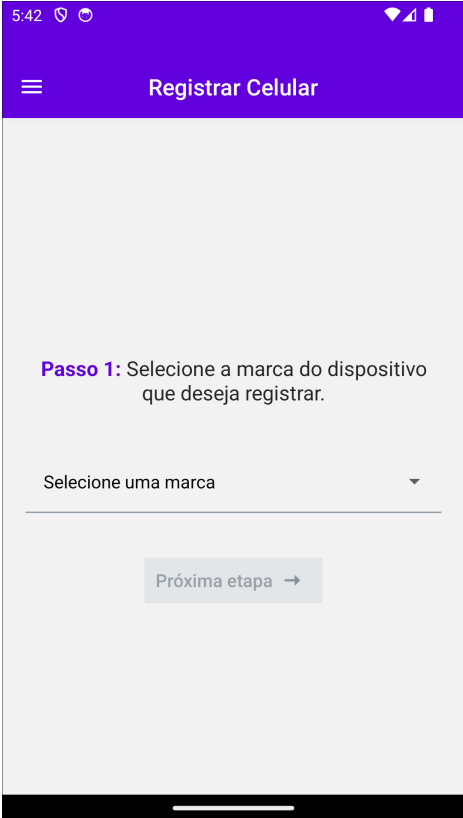


Fonte: Autoria própria (2024).

A tela de *Registrar Celular*, possibilita que o usuário registre seus aparelhos celulares através do preenchimento de um formulário de múltiplas etapas, conforme ilustrado na Figura 30. Após cadastro, o dispositivo é registrado no sistema com o status de validação pendente.

Figura 30 – Tela de registro de celular: (a) Seleção de marca, (b) Seleção de modelo, (c) Informações básicas

(a) Seleção de marca



5:42

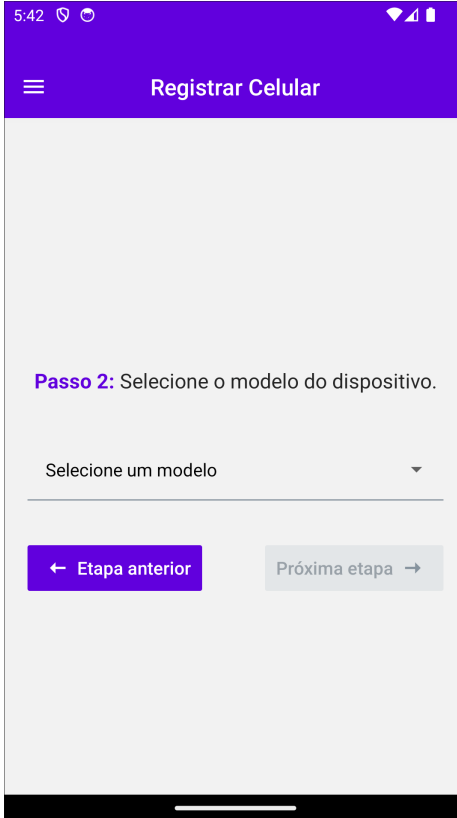
Registrar Celular

Passo 1: Selecione a marca do dispositivo que deseja registrar.

Selecione uma marca

Próxima etapa →

(b) Seleção de modelo



5:42

Registrar Celular

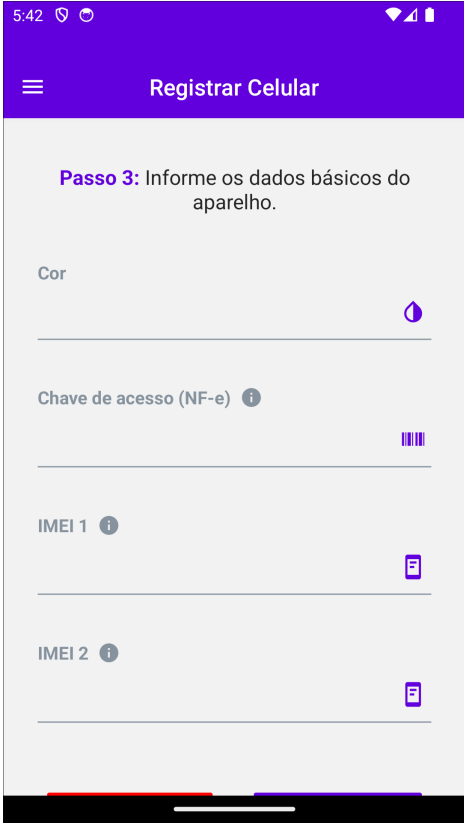
Passo 2: Selecione o modelo do dispositivo.

Selecione um modelo

← Etapa anterior

Próxima etapa →

(c) Informações básicas



5:42

Registrar Celular

Passo 3: Informe os dados básicos do aparelho.

Cor

Chave de acesso (NF-e) ⓘ

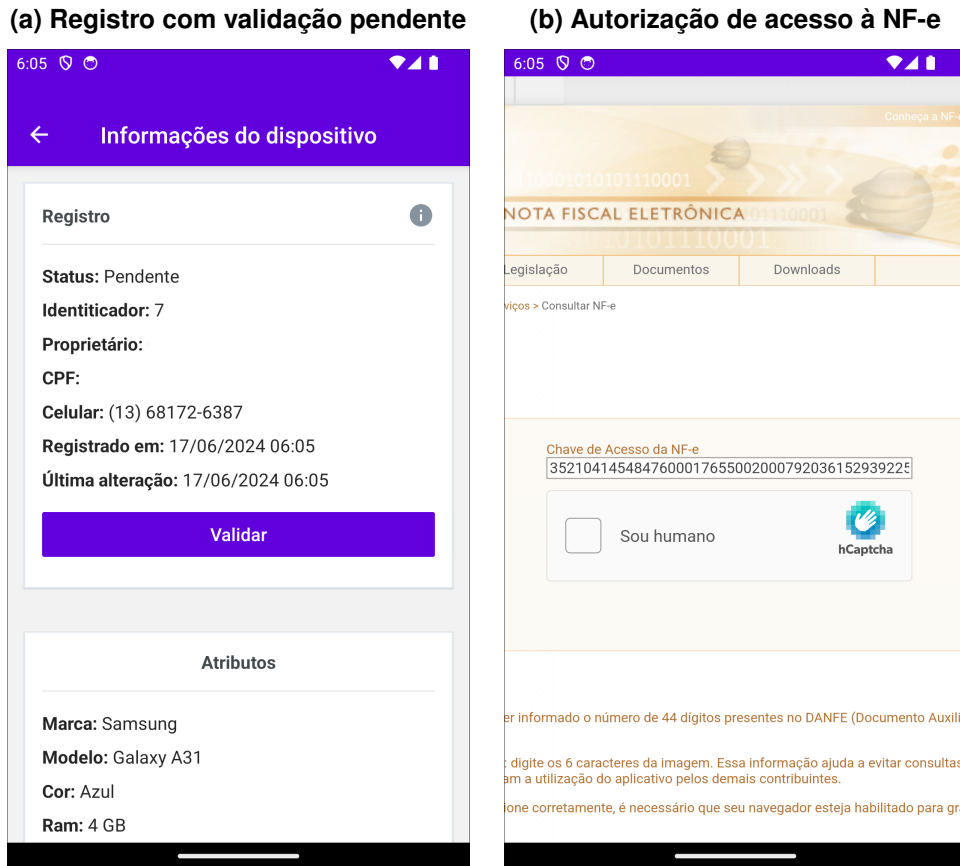
IMEI 1 ⓘ

IMEI 2 ⓘ

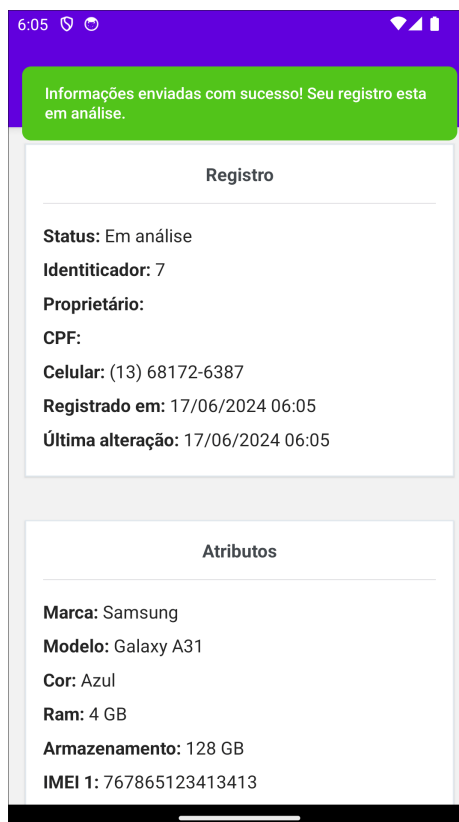
Fonte: Autoria própria (2024).

Para ter acesso a todas as funcionalidades associadas ao registro, o usuário deve realizar o processo de validação, conforme ilustrado na Figura 31. Na visualização do registro que deseja validar, o usuário deve clicar no botão “Validar”, aguardar o carregamento da página da nota fiscal eletrônica e, em seguida, clicar na opção “Sou humano” para resolver um quebra-cabeça do hCaptcha. Com esta ação, o usuário concede ao aplicativo acesso aos dados digitais da sua NF-e.

Figura 31 – Telas de validação de registro de celular: (a) Registro com validação pendente, (b) Autorização de acesso à NF-e, (c) Registro em análise após captura de dados



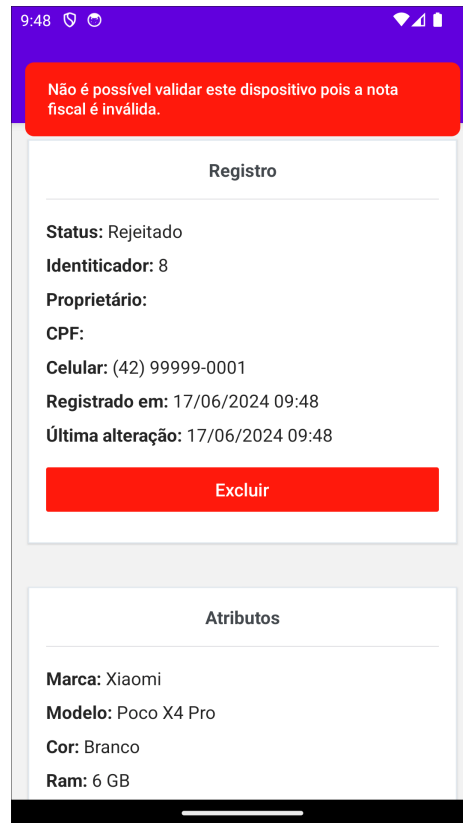
(c) Registro em análise após captura de dados de dados



Fonte: Autoria própria (2024).

Nos casos em que o usuário informa um código de acesso à NF-e inválido ou não realiza o processo de validação corretamente, este é automaticamente invalidado, conforme ilustrado na Figura 32.

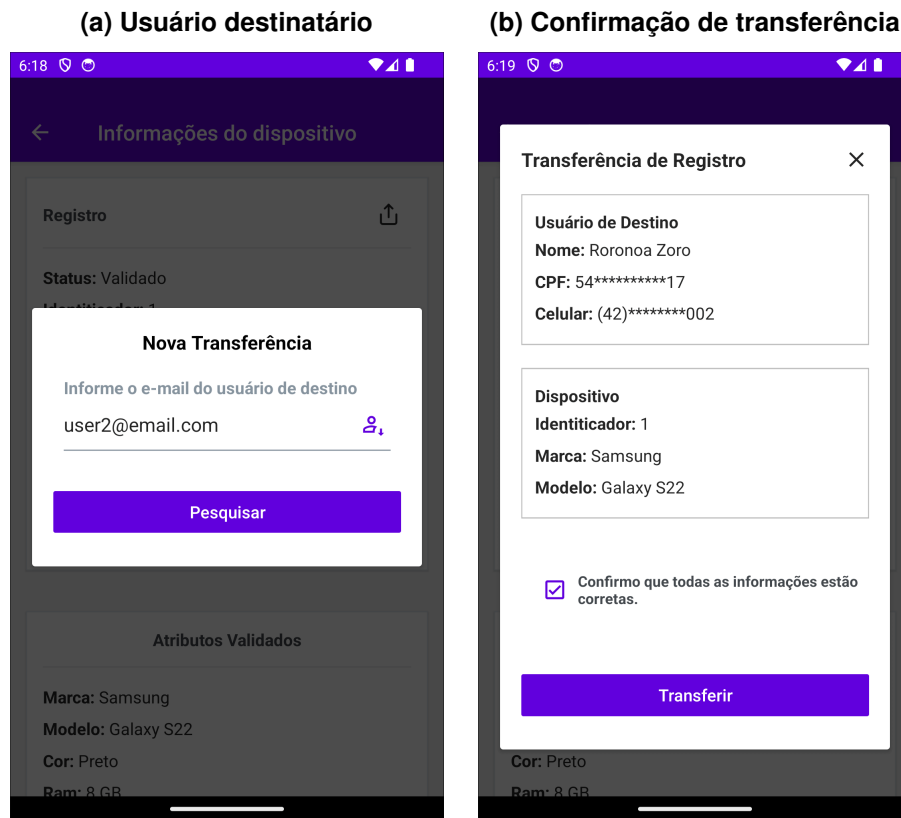
Figura 32 – Tela de registro inválido



Fonte: Autoria própria (2024).

Outro recurso para registros validados é a criação de proposta de transferência de registro. Onde o usuário proprietário do dispositivo pode criar uma transferência para qualquer outro usuário do sistema, informando o e-mail do usuário destinatário e confirmando a transferência, conforme ilustrado na Figura 33.

Figura 33 – Telas de criação de transferência: (a) Usuário destinatário, (b) Confirmação de transferência

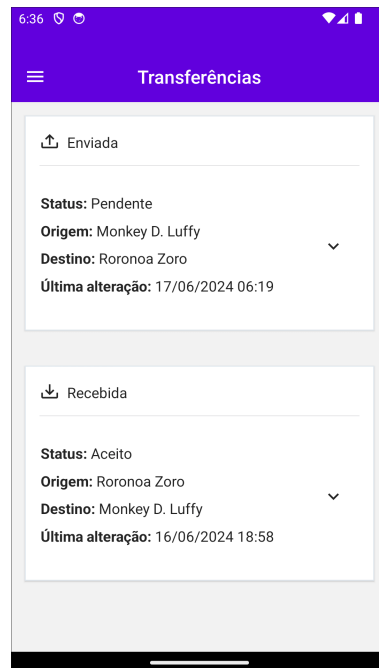


Fonte: Autoria própria (2024).

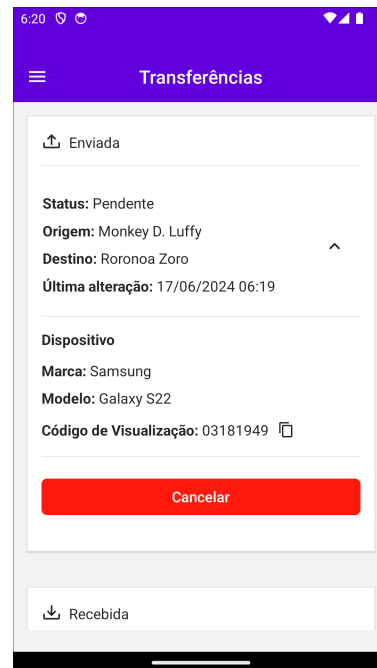
Por fim, temos a tela de *Transferências* onde são listadas todas as transferências que o usuário enviou ou recebeu. A Figura 34, ilustra os detalhes das transferências de dispositivos entre usuários.

Figura 34 – Tela de transferências: (a) Listagem de Transferências, (b) Transferência enviada, (c) Transferência recebida

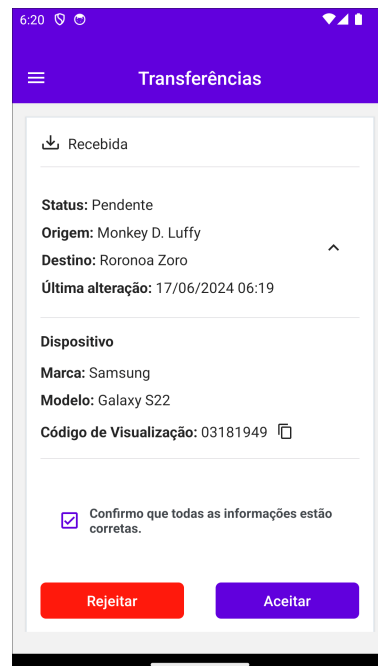
(a) Listagem de Transferências



(b) Transferência enviada



(c) Transferência recebida



Fonte: Autoria própria (2024).

6 CONCLUSÃO

Este trabalho propôs o desenvolvimento de um sistema para rastreabilidade de celulares, visando auxiliar os usuários na verificação da procedência de aparelhos no mercado de seminovos. Através da criação de uma API e de um aplicativo móvel, o projeto buscou oferecer uma plataforma segura e intuitiva para o gerenciamento de registros de celulares, utilizando a NF-e como elemento central no processo de validação.

Os resultados obtidos com o desenvolvimento do sistema demonstram que todos os objetivos propostos na seção 1.1 foram alcançados, atendendo às expectativas iniciais. Um ponto a ser discutido é a confiabilidade dos dados extraídos da NF-e, pois mesmo que este processo seja automatizado no aplicativo móvel, ele pode ser vulnerável a imprecisões ou fraudes.

Em suma, o desenvolvimento deste trabalho explorou diferentes possibilidades para solucionar o problema levantado no decorrer da pesquisa, buscando a melhor alternativa entre ferramentas e técnicas. A solução final, apesar de algumas limitações, demonstra o potencial do sistema em auxiliar os usuários a obter informações sobre a procedência de um aparelho celular, contribuindo para que tomem decisões mais seguras e, por meio da informação, evitem golpes relacionados à compra destes dispositivos.

6.1 Trabalhos futuros

Apesar de terem sido alcançados os objetivos propostos, alguns requisitos funcionais, sendo eles: **RF004**, **RF011**, **RF012** e **RF013**, não foram implementados e requerem desenvolvimento futuro para agregar mais qualidade e melhorias contínuas ao sistema.

Devido ao cronograma deste trabalho, não foi possível realizar um estudo aprofundado sobre as implicações e vulnerabilidades no processo de extração de dados da NF-e pelo aplicativo móvel. Futuramente, é necessário explorar os riscos dessa abordagem e buscar novas alternativas para melhorar a segurança na extração de dados para validação do registro.

Outro ponto importante é configurar a aplicação em ambiente de produção, possibilitando que usuários reais tenham acesso aos recursos desenvolvidos. Com registros verdadeiros, será possível rever o processo de validação do celular e refinar os valores das constantes de similaridade mínima exigida para cada atributo durante a validação.

Por fim, é importante expandir a base de dados de dispositivos suportados pelo sistema, permitindo que mais marcas e modelos sejam atendidos, aumentando assim o alcance e a utilidade do sistema.

REFERÊNCIAS

- ANATEL, A. N. de T. Relatório anual - exercício 1999, da agência nacional de telecomunicações. 1999. Disponível em: <https://sistemas.anatel.gov.br/anexar-api/publico/portal-publicar/documentos?numeroPublicacao=19795>. Acesso em: 21 Ago. 2022.
- ANATEL, A. N. de T. Relatório anual - exercício 2003, da agência nacional de telecomunicações. 2003. Disponível em: <https://sistemas.anatel.gov.br/anexar-api/publico/portal-publicar/documentos?numeroPublicacao=106090>. Acesso em: 21 Ago. 2022.
- ANATEL, A. N. de T. Relatório anual - exercício 2007, da agência nacional de telecomunicações. 2007. Disponível em: <https://sistemas.anatel.gov.br/anexar-api/publico/portal-publicar/documentos?numeroPublicacao=263757>. Acesso em: 21 Ago. 2022.
- ANATEL, A. N. de T. **Celular Legal**. 2022. Disponível em: <https://www.gov.br/anatel/pt-br/assuntos/celular-legal>. Acesso em: 8 Set. 2022.
- ANNIE, A. **State of App Marketing Brazil - Edição de 2021**. 2021. Disponível em: <https://dataai.infogram.com/pt-brazil-state-of-app-marketing-brazil-2021-1h7k230dg18pg2x>. Acesso em: 16 Set. 2022.
- BASTOS, F. **Roubo de celular: veja 5 dicas para proteger informações financeiras**. 2022. Disponível em: <https://exame.com/invest/minhas-financas/roubo-celular-5-dicas-para-se-proteger/>. Acesso em: 16 Set. 2022.
- BATHULA, C. P. **Understanding Fuzzy String Matching: Exploring Fuzz Ratio, Fuzz Partial Ratio, Token Set Ratio, and Token Sort Ratio: Machine Learning Concept 78**. 2023. Disponível em: <https://tinyurl.com/y98yv9yj>. Acesso em: 08 Jun. 2024.
- BITBUCKET. **Git Feature Branch Workflow**. 2022. Disponível em: <https://www.atlassian.com/git/tutorials/comparing-workflows/feature-branch-workflow>. Acesso em: 10 Out. 2022.
- BRASIL, R. F. do. **Portal da Nota Fiscal Eletrônica**. 2022. Disponível em: <https://www.nfe.fazenda.gov.br/portal/consultaRecaptcha.aspx?tipoConsulta=resumo&tipoConteudo=7PhJ+gAVw2g=>. Acesso em: 22 Out. 2022.
- CARRO, O. no. **Olho no Carro**. 2022. Disponível em: <https://www.olhonocarro.com.br/>. Acesso em: 5 Set. 2022.
- CNDL, C. N. de D. L. Consumo online, abril 2021. 2021. Disponível em: https://d335luupugsy2.cloudfront.net/cms/files/227307/1628710783Apresentao__release.pdf. Acesso em: 23 Ago. 2022.
- CUNHA, A. **React Native: o que é e tudo sobre o Framework**. 2022. Disponível em: <https://tinyurl.com/5cd55znd>. Acesso em: 15 Nov. 2022.
- EXPO. **Make any app. Run it everywhere**. 2022. Disponível em: <https://expo.dev/>. Acesso em: 15 Nov. 2022.
- FBSP, F. brasileiro de segurança pública. Anuário brasileiro de segurança pública 2022. 2022. Disponível em: <https://forumseguranca.org.br/wp-content/uploads/2022/06/anuario-2022.pdf?v=5>. Acesso em: 23 Ago. 2022.
- GIT. **About**. 2022. Disponível em: <https://git-scm.com/about>. Acesso em: 20 Nov. 2022.

- GITHUB. **Documentação do GitHub Actions**. 2022. Disponível em: <https://docs.github.com/pt/actions>. Acesso em: 20 Nov. 2022.
- IBM. **O que é o Docker?** 2024. Disponível em: <https://www.ibm.com/br-pt/topics/docker>. Acesso em: 27 Mai. 2024.
- LARAVEL. **Database: Getting Started**. 2024. Disponível em: <https://laravel.com/docs/11.x/database>. Acesso em: 26 Mai. 2024.
- LARAVEL. **Eloquent: Getting Started**. 2024. Disponível em: <https://laravel.com/docs/11.x/eloquent>. Acesso em: 26 Mai. 2024.
- LARAVEL. **Laravel Sail**. 2024. Disponível em: <https://laravel.com/docs/11.x/sail>. Acesso em: 30 Mai. 2024.
- LARAVEL. **The PHP Framework for Web Artisans**. 2024. Disponível em: <https://laravel.com/>. Acesso em: 26 Mai. 2024.
- MEIRELLES, F. de S. 34ª pesquisa anual do fgvcia: Uso da ti nas empresas. 2023. Disponível em: https://eaesp.fgv.br/sites/eaesp.fgv.br/files/u68/pesti-fgvcia-2023_0.pdf. Acesso em: 22 Mai. 2024.
- MJSP, M. da Justiça e S. P. **Celular Seguro BR**. 2024. Disponível em: <https://www.gov.br/pt-br/apps/celular-seguro-br>. Acesso em: 25 Mai. 2024.
- PYKES, K. **Fuzzy String Matching in Python Tutorial**. 2023. Disponível em: <https://www.datacamp.com/tutorial/fuzzy-string-python>. Acesso em: 08 Jun. 2024.
- REGISTRADA, B. **Bike Registrada**. 2022. Disponível em: <https://bikeregistrada.com.br/>. Acesso em: 23 Ago. 2022.
- TECNOBLOG. **iPhone 11 e 7 lideram ranking de celulares usados mais vendidos no Brasil; confira o top 50**. 2023. Disponível em: <https://tecnoblog.net/noticias/iphone-11-e-7-lideram-ranking-de-celulares-usados-mais-vendidos-no-brasil-confira-o-top-50/>. Acesso em: 08 Jun. 2024.
- VIEIRA, D. **O que é MySQL? Dicas de como usar o banco de dados**. 2023. Disponível em: <https://www.hostgator.com.br/blog/mysql-e-suas-vantagens/>. Acesso em: 26 Mai. 2024.