



UNIVERSIDADE
ESTADUAL DE LONDRINA

JEAN CARLO POLO

**DESENVOLVIMENTO DE UM SISTEMA COM
BANCO DE DADOS RELACIONAL**

Londrina
2025

JEAN CARLO POLO

**DESENVOLVIMENTO DE UM SISTEMA COM
BANCO DE DADOS RELACIONAL**

Trabalho da disciplina Banco de Dados I, do curso de graduação de Ciência da Computação, da Universidade Estadual de Londrina – UEL.

Professor: Anderson Paulo Avila Santos

Londrina
2025

SUMÁRIO

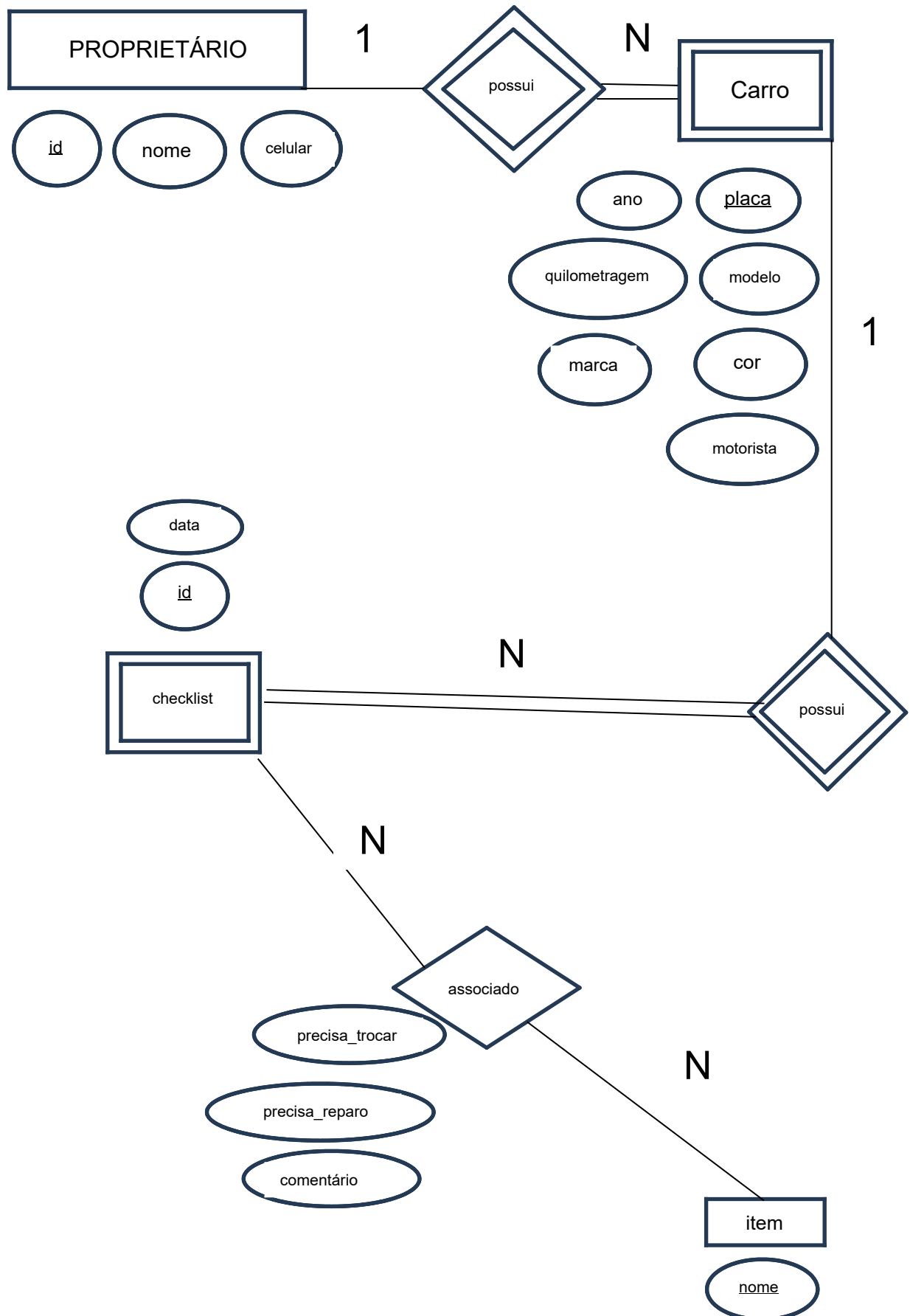
1	TEMA E JUSTIFICATIVA.....	3
2	DIAGRAMA ENTIDADE-RELACIONAMENTO (DER).....	4
3	TECNOLOGIAS UTILIZADAS	5
4	DESCRIÇÃO DAS FUNCIONALIDADES DO SISTEMA	6
5	CONSULTAS SQL UTILIZADAS	8
6	CAPTURAS DE TELA DA INTERFACE	12
	REFERÊNCIAS	20

1 TEMA E JUSTIFICATIVA

O desenvolvimento deste aplicativo justifica-se pela necessidade de implementar uma solução tecnológica que possibilite o gerenciamento eficiente e integrado dos processos de manutenção de veículos. Em um cenário onde a segurança e a performance dos automóveis são de suma importância, o sistema proposto visa consolidar informações relativas aos proprietários, veículos e registros de manutenção (checklists), facilitando a identificação precoce de eventuais problemas e permitindo a adoção de medidas preventivas.

Ao oferecer funcionalidades abrangentes – que incluem operações de cadastro, atualização e remoção de registros, bem como a geração de relatórios detalhados – o aplicativo contribui para a otimização dos custos com reparos e aumenta a confiabilidade dos dados utilizados para a tomada de decisão. Dessa forma, a solução se torna relevante tanto para a gestão individual de veículos quanto para a administração de frotas em ambientes corporativos, promovendo a melhoria contínua da manutenção preventiva e a maximização da segurança operacional.

2 DIAGRAMA ENTIDADE-RELACIONAMENTO (DER)



3 TECNOLOGIAS UTILIZADAS

No desenvolvimento da aplicação, optou-se por utilizar o framework Flutter em conjunto com a linguagem de programação orientada a objetos Dart. Essa combinação permite a criação de interfaces de usuário responsivas e de alta performance, além de possibilitar o desenvolvimento multiplataforma, atendendo a diversos sistemas operacionais com um único código-fonte.

Para o armazenamento local dos dados, empregou-se a biblioteca SQFlite, que oferece uma interface para o sistema de gerenciamento de banco de dados SQLite. Essa solução se destaca pela leveza e eficiência, sendo amplamente utilizada em aplicações móveis que requerem persistência de dados de forma segura e confiável.

No que tange ao registro e monitoramento das ações e ocorrências, o pacote Logger foi incorporado ao projeto. Essa ferramenta é fundamental para a captura de logs, permitindo a identificação e a correção de erros, bem como o acompanhamento das operações realizadas durante a execução do aplicativo.

A funcionalidade de paginação, especialmente em listas extensas como as de proprietários e carros, foi implementada por meio da biblioteca Infinite Scroll Pagination. Essa tecnologia possibilita a exibição gradual dos itens, otimizando o desempenho da aplicação e melhorando a experiência do usuário ao evitar o carregamento de grandes volumes de dados de uma única vez.

Adicionalmente, para viabilizar a implementação de ações deslizantes, como a exclusão e edição de itens, foi utilizada a biblioteca Flutter Slidable. Esse recurso proporciona uma interface interativa e intuitiva, permitindo que o usuário realize modificações de forma ágil e prática.

Por fim, o gerenciamento de eventos e a atualização dinâmica das interfaces foram assegurados pela utilização da biblioteca Signals Flutter. Essa ferramenta facilita a comunicação entre os diversos componentes da aplicação, garantindo que as alterações no estado dos dados sejam refletidas em tempo real na interface do usuário.

4 DESCRIÇÃO DAS FUNCIONALIDADES DO SISTEMA

O sistema implementa um conjunto de operações para o gerenciamento de entidades relacionadas à administração de veículos e seus respectivos proprietários, bem como para o controle das manutenções realizadas. Inicialmente, dispõe-se de um módulo destinado ao gerenciamento dos proprietários, permitindo a realização de operações de criação, leitura, atualização e exclusão (CRUD). Esse módulo possibilita o cadastro detalhado dos responsáveis pelos veículos, garantindo a integridade e a rastreabilidade dos dados referentes à titularidade.

Complementarmente, o aplicativo incorpora um módulo específico para o gerenciamento dos automóveis, no qual cada proprietário pode estar associado a múltiplos veículos. Nesse contexto, são disponibilizadas as funcionalidades necessárias para o cadastro, a atualização e a remoção de registros, assegurando uma relação consistente e hierarquizada entre os proprietários e seus carros. Cada veículo também possui uma cor conforme o código de cores de veículos novos da NF-e (Nota Fiscal Eletrônica), o cCorDENATRAN¹.

No que concerne à manutenção e controle das condições dos veículos, o sistema implementa um módulo de checklists. Esse componente permite a criação, atualização e exclusão de registros de checklists, cada um associado a um veículo específico, de forma a registrar as inspeções realizadas e os procedimentos de manutenção recomendados.

Adicionalmente, cada checklist é composto por diversos itens, os quais podem ser individualmente avaliados. Esses itens oferecem a possibilidade de serem marcados para indicar a necessidade de reparo ou substituição, além de permitir a inclusão de observações complementares que detalhem as condições de cada componente ou sistema avaliado.

Por fim, o aplicativo dispõe de uma funcionalidade de geração de relatórios, os quais são copiados para a área de transferência (clipboard) do sistema. Esses

¹ <https://atendimento.tecnospeed.com.br/hc/pt-br/articles/360015071153-Regra-Tributaria-cCorDENATRAN-C%C3%B3digo-da-Cor-Ve%C3%ADculos-novos-da-NF-e>

relatórios contêm informações consolidadas provenientes do banco de dados, proporcionando uma visão abrangente do estado dos registros e permitindo uma análise rápida e eficiente dos dados armazenados.

5 CONSULTAS SQL UTILIZADAS

As consultas SQL utilizadas no aplicativo se encaixam em três categorias:

1ª. Verificação de Existência

Verificar se um elemento de uma tabela existe (na inserção de um elemento dependente):

```
SELECT EXISTS(SELECT 1 FROM tabela WHERE chavePrimária = ?)
```

2ª Listagem e Busca de Registros

Listar elementos de uma tabela com paginação

```
SELECT * FROM tabela LIMIT 10 OFFSET ?
```

Listar todos os elementos da tabela

```
SELECT * FROM tabela
```

Buscar um elemento pela chave primária

```
SELECT * FROM tabela WHERE chavePrimária = ?
```

Listar os Ids de itens já associados ao checklist

```
SELECT itemId FROM checklistItem WHERE checklistId = ?
```

Listar todos os detalhes dos itens associados ao checklist

```
SELECT item.id, precisaReparo, precisaTrocar, observacao, checklistId, itemId  
FROM item
```

```
LEFT JOIN checklistItem
```

```
    ON item.id = checklistItem.itemId
```

```
    AND checklistItem.checklistId = ?
```

3ª Relatórios

Busca os carros que, na sua checklist mais recente, possuem algum item que precise de reparo ou troca

```

SELECT
    p.nome AS 'Dono',
    c.placa AS 'Placa do Carro',
    SUM(CASE WHEN ci.precisaReparo = 1 OR ci.precisaTrocar = 1 THEN 1 ELSE 0
          END) AS 'Quantidade de Problemas'
FROM checklistItem ci
INNER JOIN checklist ch ON ci.checklistId = ch.id
INNER JOIN (
    -- Subconsulta para obter o checklist mais recente de cada carro
    SELECT placa, MAX(dataHorario) AS ultimo_dataHorario
    FROM checklist
    GROUP BY placa
) ultima_checklist ON ch.placa = ultima_checklist.placa AND ch.dataHorario =
    ultima_checklist.ultimo_dataHorario
INNER JOIN carro c ON ch.placa = c.placa
INNER JOIN proprietario p ON c.proprietarioId = p.id
GROUP BY c.placa, p.nome
HAVING 'Quantidade de Problemas' > 0
ORDER BY SUM(CASE WHEN ci.precisaReparo = 1 OR ci.precisaTrocar = 1 THEN
    1 ELSE 0 END) DESC;

```

checklistItem e checklist: Associa os itens de checklist ao checklist correspondente.

Subconsulta ultima_checklist: Obtém, para cada carro (identificado pela placa), a data/hora máxima, ou seja, a última checklist registrada.

Junção com carro e proprietario: Relaciona o carro com seu proprietário.

O HAVING 'Quantidade de Problemas' > 0 garante que somente serão exibidos os carros cuja última checklist apresente pelo menos um problema.

Ordena os resultados de forma decrescente com base na quantidade de problemas, de modo que os carros com mais problemas apareçam primeiro.

Agrupa os registros da tabela carro por proprietário e conta quantos carros cada um possui

```

SELECT

```

```

p.nome AS 'Dono',
COUNT(c.placa) AS 'Quantidade de Carros'
FROM carro c
INNER JOIN proprietario p ON c.proprietariold = p.id
GROUP BY p.nome
ORDER BY COUNT(c.placa) DESC;

```

Associa cada carro ao seu respectivo dono através do campo proprietariold.

Usa GROUP BY p.nome para agrupar os carros de acordo com o nome do proprietário.

COUNT(c.placa) conta o número de carros para cada dono.

A cláusula ORDER BY COUNT(c.placa) DESC ordena os resultados do proprietário com mais carros para o que tem menos.

Determina, para cada carro, há quanto tempo foi registrada a última checklist (em segundos). Caso o carro nunca tenha tido uma checklist, utiliza o tempo atual (ou seja, segundos desde meia-noite de 01 de janeiro de 1970).

```

SELECT

```

```

p.nome AS "Nome do Dono",
c.placa AS "Placa do Carro",
COALESCE(strftime('%s','now') - MAX(ch.dataHorario), strftime('%s','now')) AS
    "Tempo Desde Última Checklist (segundos)"

```

```

FROM carro c

```

```

LEFT JOIN checklist ch ON c.placa = ch.placa

```

```

LEFT JOIN proprietario p ON c.proprietariold = p.id

```

```

GROUP BY c.placa

```

```

ORDER BY "Tempo Desde Última Checklist (segundos)" DESC;

```

LEFT JOIN checklist: Liga os carros aos seus checklists (caso existam). O uso do LEFT JOIN garante que todos os carros serão retornados, mesmo que não tenham checklist.

LEFT JOIN proprietario: Associa o carro ao seu dono.

Utiliza strftime('%s','now') para obter o timestamp atual em segundos.

Subtrai MAX(ch.dataHorario) (a última data/hora registrada no checklist para aquele carro) para calcular quantos segundos se passaram.

COALESCE: Se não houver nenhuma checklist (ou seja, MAX(ch.dataHorario) é nulo), retorna o próprio valor de strftime('%s','now').

Agrupar os resultados por c.placa para obter o último registro por carro.

Ordenar os carros de forma decrescente com base no tempo decorrido, de modo que os carros que estão há mais tempo sem uma checklist apareçam primeiro.

6 CAPTURAS DE TELA DA INTERFACE

Figura 1
CRUD - Proprietários

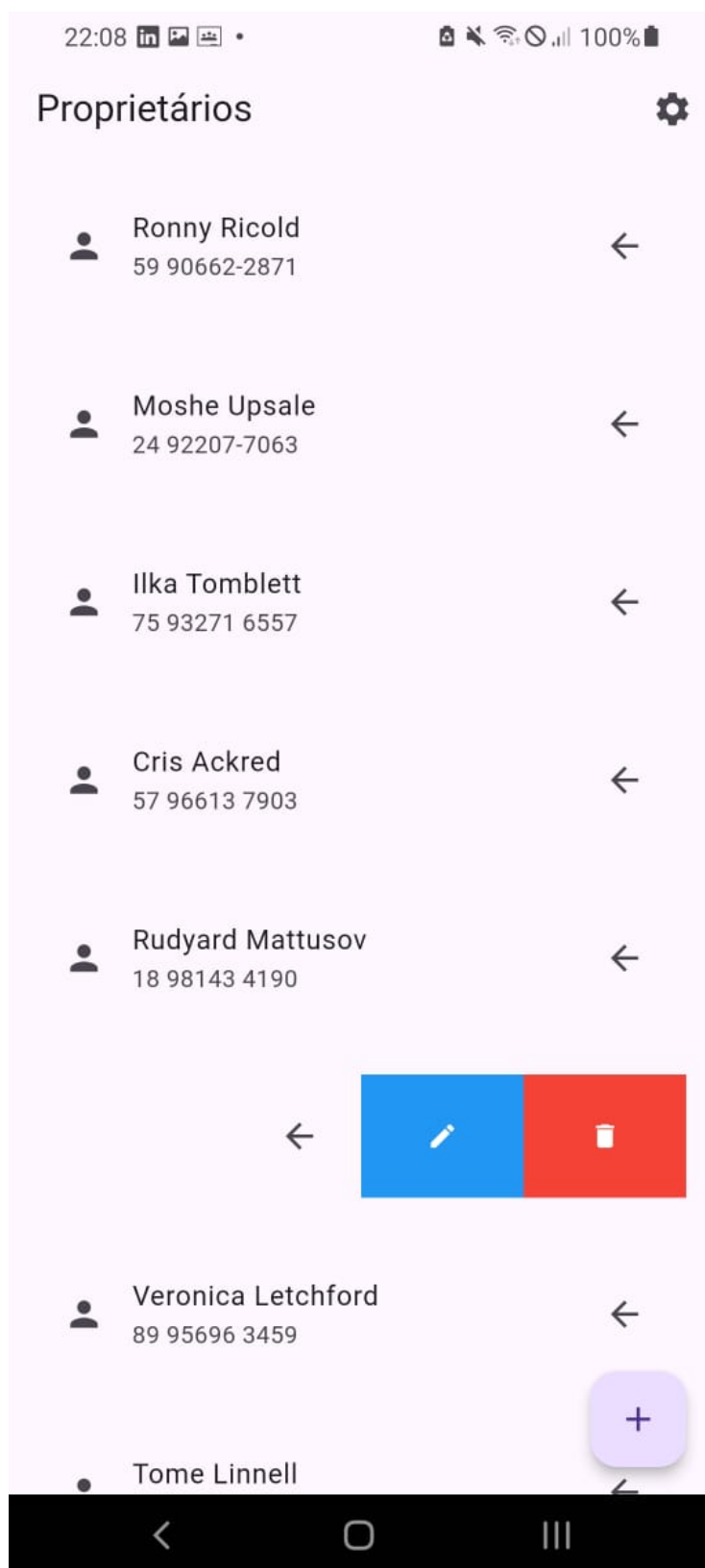


Figura 2
Criação de Proprietário

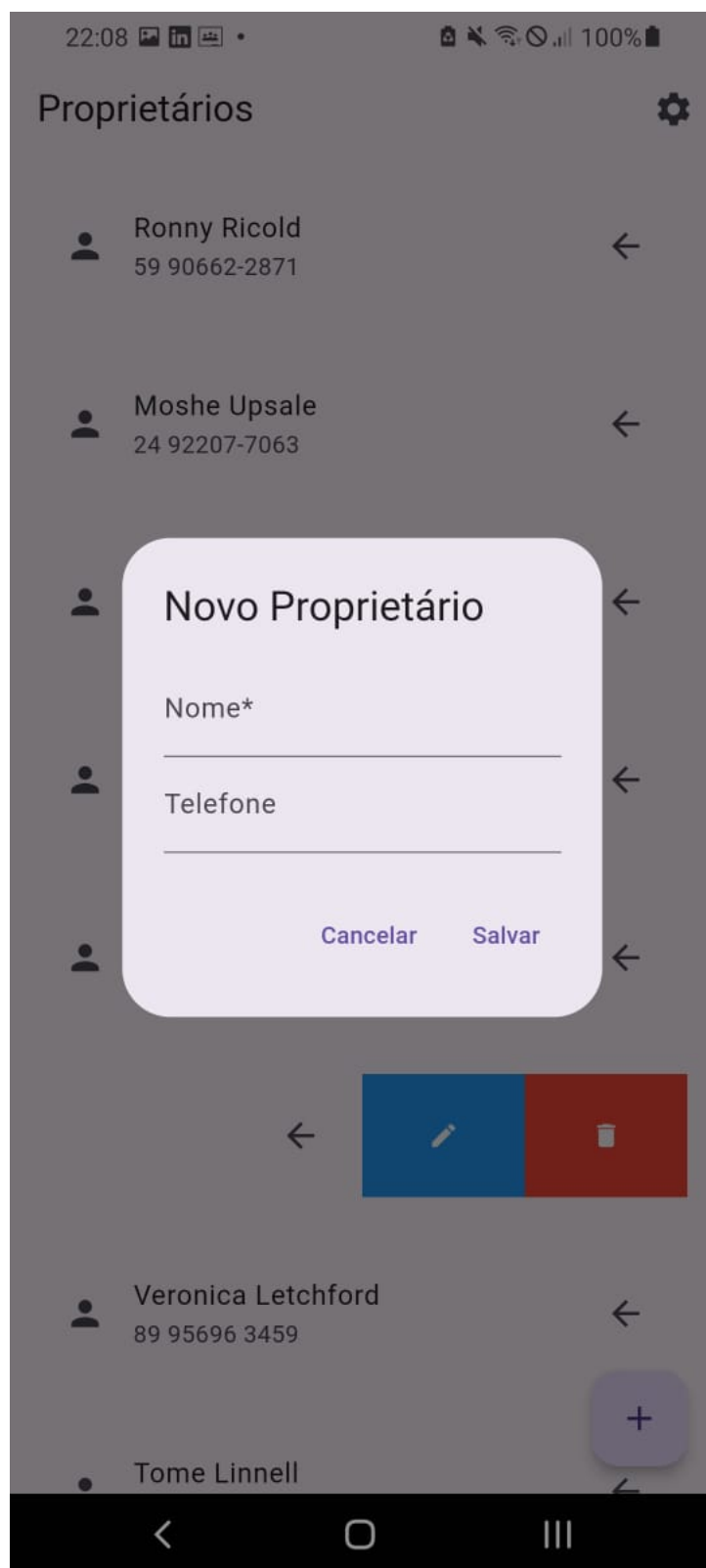


Figura 3
Configurações

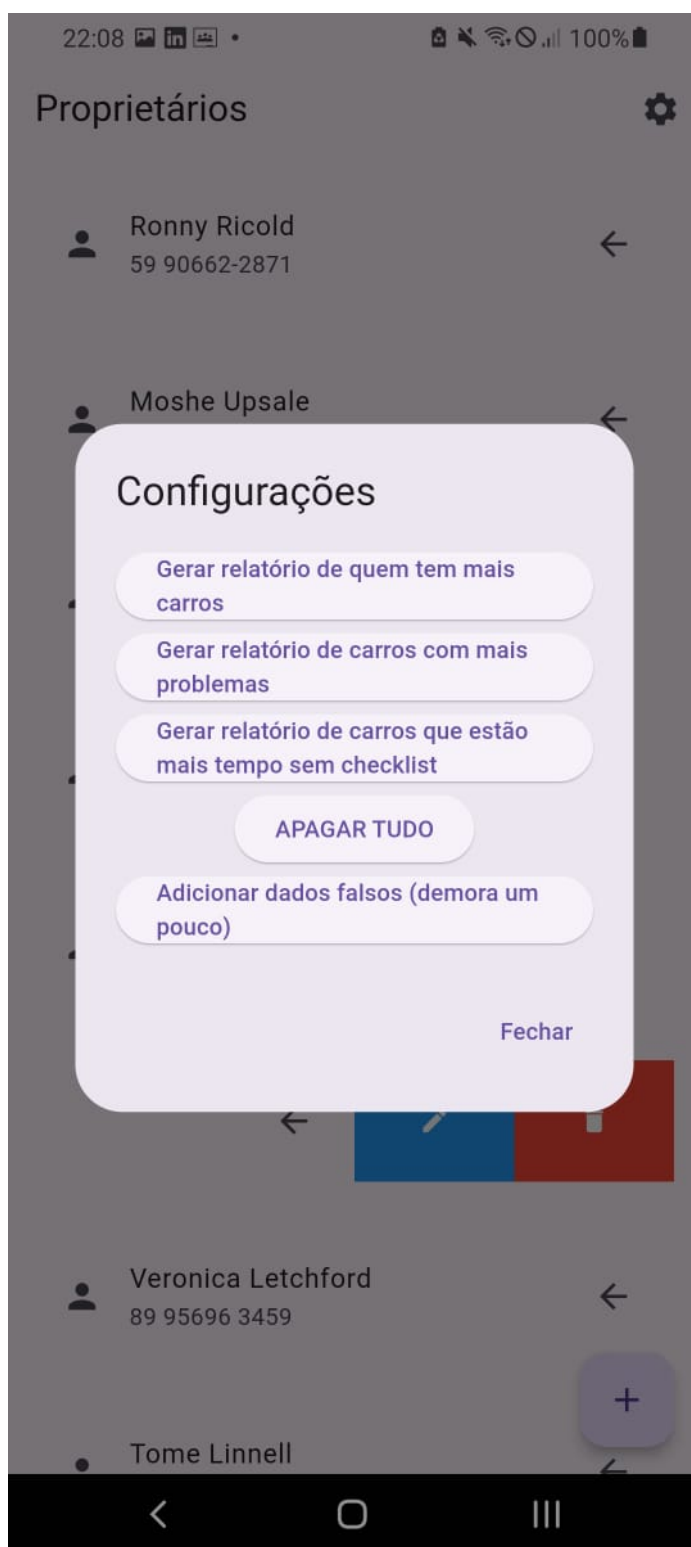


Figura 4
CRUD - Carros

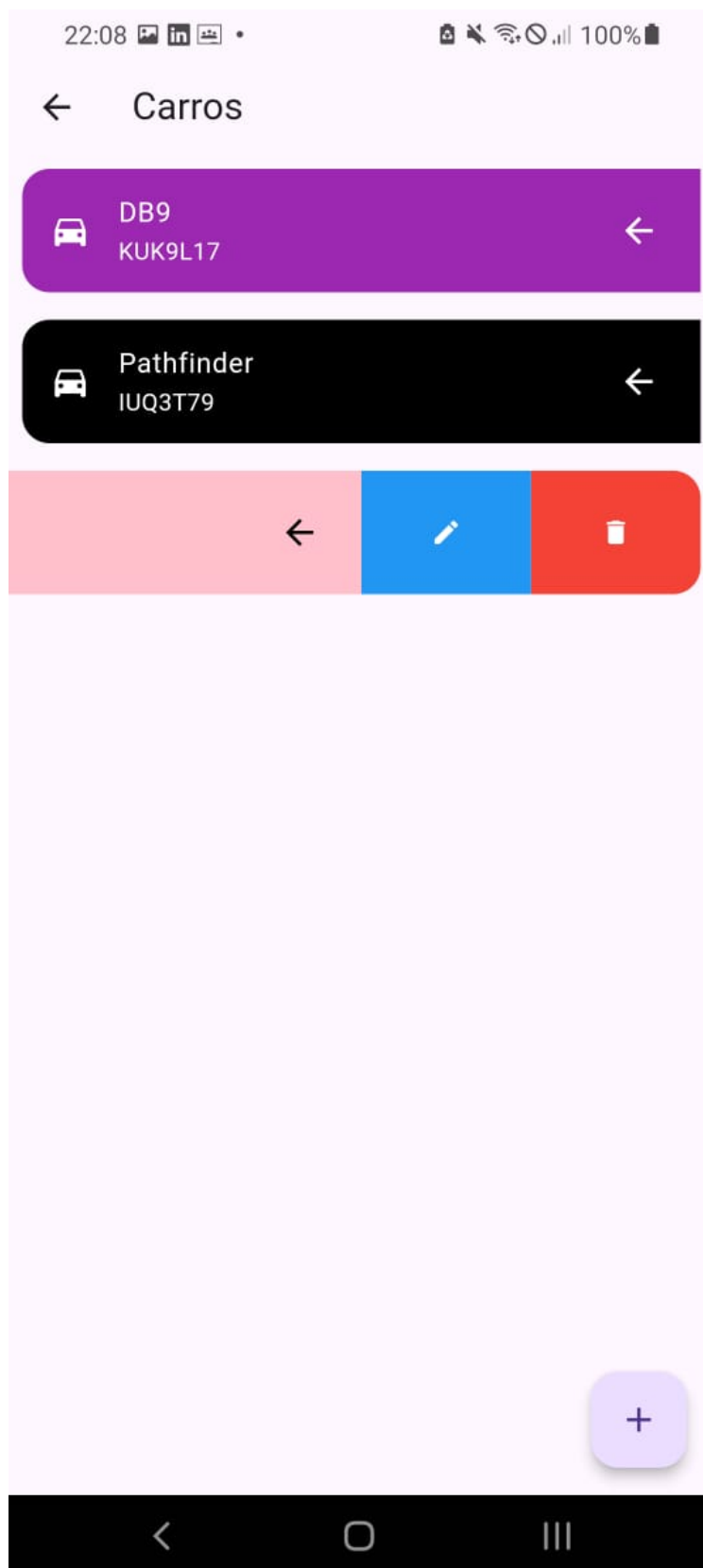


Figura 5
Criação de Carro

The image shows a mobile application interface for managing cars. In the background, there is a list titled "Carros" with a back arrow. The first item in the list is a purple card with a car icon, the text "DB9 KUK9L17", and a right-pointing arrow. Below it is a black card with a car icon and a right-pointing arrow. A white modal form titled "Novo Carro" is overlaid on the screen. The form contains the following fields: "Placa*" (empty), "Modelo" (empty), "Cor" (a dropdown menu showing a white color swatch and the text "Branca"), "Motorista" (empty), "Marca" (empty), "Ano" (with the value "2025" entered), and "Quilometragem" (with the value "0" entered). At the bottom of the form are two buttons: "Cancelar" and "Salvar". In the bottom right corner of the app screen, there is a grey rounded square button with a white plus sign. The Android navigation bar is visible at the very bottom.

22:08 100%

← Carros

DB9 KUK9L17

Novo Carro

Placa*

Modelo

Cor

Branca

Motorista

Marca

Ano

2025

Quilometragem

0

Cancelar Salvar

+

Figura 6
Seletor de Cor para o Carro

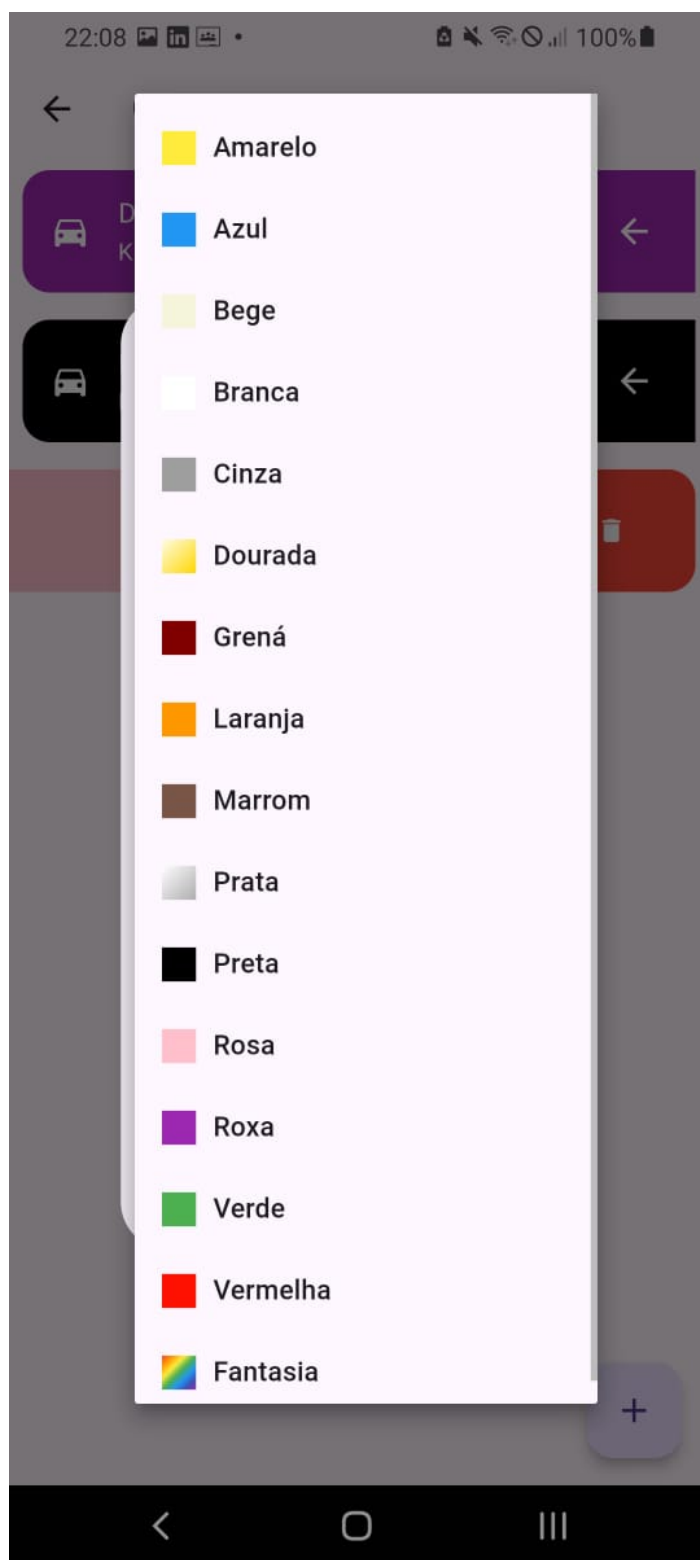


Figura 7
CRUD - Checklists



Figura 8
Itens da Checklist



BIBLIOGRAFIA

GUELF, Everton. Regra Tributária – COORDENATRAN – Código da cor (veículos novos) da NF-E. Disponível em: <https://atendimento.tecnospeed.com.br/hc/pt-br/articles/360015071153-Regra-Tributaria-cCorDENATRAN-C%C3%B3digo-da-Cor-Ve%C3%ADculos-novos-da-NF-e> Acesso em: 14 fev. 2025.