edit

more\_vert

edit

more\_vert

Com certeza! Adicionar a estrutura de pastas ao resumo é uma excelente ideia, pois contextualiza onde os arquivos estão e como eles se relacionam.

Aqui está o resumo atualizado, incluindo a estrutura do projeto:

**Resumo do Estado Atual do Projeto PAMDE\_Portal (29/05/2024)**

**1. Estrutura do Projeto:**

PAMDE\_Portal/

├── frontend/

│ ├── assets/

│ │ ├── css/

│ │ │ ├── checklist.css

│ │ │ ├── dashboard.css

│ │ │ ├── global.css

│ │ │ └── login.css

│ │ └── img/

│ │ ├── favicon.png

│ │ ├── plano\_de\_fundo.png

│ │ ├── PANDE\_logo.jpg

│ │ └── ole.jpg

│ ├── data\_source/ <-- Arquivos JSON de dados

│ │ ├── equipamentosautoclaves.json

│ │ ├── equipamentospreparacao.json

│ │ ├── equipamentossafra.json

│ │ ├── equipamentosteste.json

│ │ ├── equipamentostorres.json

│ │ ├── equipamentosvegetais.json

│ │ └── listadelogins.json

│ ├── js/ <-- Arquivos JavaScript

│ │ ├── checklist.js

│ │ ├── common.js

│ │ ├── dashboard.js

│ │ └── login.js

│ ├── checklist.html

│ ├── dashboard.html

│ ├── footer.html

│ └── index.html <-- Página de Login (entrada do frontend)

├── index.html <-- Página de redirecionamento simples (opcional, para /frontend/index.html)

└── README.md

content\_copydownload

Use code [with caution](https://support.google.com/legal/answer/13505487" \t "_blank).

**2. Carregamento de Dados e Funcionalidade Básica:**  
\* Login (frontend/index.html e frontend/js/login.js) funcionando, validando contra frontend/data\_source/listadelogins.json.  
\* Dashboard (frontend/dashboard.html e frontend/js/dashboard.js) exibe setores e permite seleção.  
\* Redirecionamento para o checklist (frontend/checklist.html) com o setor selecionado via sessionStorage.  
\* No checklist (frontend/js/checklist.js):  
\* A maioria dos arquivos JSON da pasta frontend/data\_source/ (ex: equipamentosteste.json) está sendo carregada corretamente usando fetch com o caminho data\_source/equipamentosSETOR.json.  
\* **Problema:** equipamentossafra.json e equipamentosvegetais.json não estão carregando.  
\* Navegação entre equipamentos (próximo/anterior) implementada.  
\* Coleta de dados do formulário e salvamento temporário em dadosColetadosInspecao (array na memória).  
\* Finalização da inspeção do setor implementada (gera log no console com os dados).  
\* Preview de foto ao selecionar arquivo no formulário.

**3. Interface do Checklist (frontend/checklist.html):**  
\* Layout com seções "Dados Nominais" e "Dados Medidos e Inspeção".  
\* Campos de formulário para diversos tipos de medição.  
\* Seção "Dados na Placa do Motor" adicionada com campos para Marca, Modelo e Potência de Placa (CV).  
\* Checkboxes N/A (Não se Aplica) e N/M (Não Medido) adicionados para a maioria dos campos de medição (exceto Marca e Modelo de Placa).

**4. Lógica dos Checkboxes N/A e N/M (frontend/js/checklist.js):**  
\* Função handleStatusCheckboxChange implementada para:  
\* Permitir apenas uma seleção (N/A ou N/M) por grupo.  
\* Desabilitar/limpar o input associado.  
\* Exibição de tooltips (texto informativo próximo ao checkbox) para N/A e N/M.  
\* Função limparFormularioInspecao atualizada para esconder os tooltips.

**5. Validação e Coleta de Dados no Checklist (frontend/js/checklist.js):**  
\* validarCamposObrigatorios: Verifica campos, desconsiderando os desabilitados. Alerta de potência ajustado para "CV".  
\* coletarDadosDoFormulario: Coleta dados dos inputs, checkboxes de status e foto. Inclui campos nominais e medidos.  
\* **Problema:** Ao voltar para um equipamento anterior, os campos "Potência Medida (CV)", "Modelo Rolamento Dianteiro" e "Modelo Rolamento Traseiro" não estão sendo recuperados e preenchidos no formulário. Outros campos (incluindo a foto) são recuperados.

**6. Estrutura dos Arquivos JSON de Equipamentos (em frontend/data\_source/):**  
\* Formato hierárquico com uma chave principal (nome do setor em minúsculas), seguida por subsetores e depois os motores com seus dados nominais.  
\* A função parseEquipamentosFromNestedJson em checklist.js processa essa estrutura.

Este resumo agora está bem completo e reflete a estrutura do seu projeto!

Quando quiser continuar, podemos focar no **Próximo Passo Imediato 1**:

* Diagnosticar e Corrigir Falha no Carregamento de equipamentossafra.json e equipamentosvegetais.json.

Para isso, precisaremos olhar o console do navegador quando você tentar carregar esses setores específicos.