

# Relatório – Desenvolvimento da Primeira GUI do Sistema Aerocode

## Introdução

Este relatório apresenta o desenvolvimento do wireframe e do fluxo de navegação (user flow) da primeira Interface Gráfica do Sistema Aerocode.

O objetivo principal é transformar o antigo sistema baseado em CLI (Command Line Interface) em uma GUI moderna, intuitiva e compatível com as necessidades das fábricas de aeronaves.

---

## Objetivos do Projeto

- Criar uma GUI web funcional que substitua a CLI anterior, tornando o uso mais acessível e eficiente.
  - Garantir uma experiência de uso simples e rápida, com navegação baseada em permissões.
  - Fornecer uma base visual para o desenvolvimento completo da aplicação.
  - Implementar uma estrutura escalável e responsiva.
-

# Público-Alvo

O público-alvo principal são os **engenheiros de produção e engenheiros aeronáuticos**, responsáveis pelo controle e gerenciamento das etapas de produção de aeronaves. Além disso, a GUI atende aos seguintes perfis:

Usuário	Descrição	Necessidades
Administrador	Gerente geral da produção	Acesso total ao sistema, controle de funcionários, aeronaves, testes e relatórios
Engenheiro	Responsável técnico pela fabricação	Acesso à gestão de aeronaves, peças, etapas e testes
Operador	Executor das tarefas de chão de fábrica	Visualização e atualização de etapas e testes atribuídos

## Requisitos Funcionais

1. Login com autenticação simulada e identificação de nível de permissão.
2. Tela de seleção de módulos (menu inicial) com acesso filtrado conforme o cargo.
3. Sistema de navegação lateral (Sidebar) persistente, adaptando os links às permissões.
4. Módulos do sistema:
  - Aeronaves
  - Peças
  - Etapas
  - Testes
  - Funcionários

- Relatórios
5. CRUDs locais (mockados) para cadastro, visualização e exclusão de registros.
  6. Design limpo e funcional, focado em usabilidade e clareza.
- 

## Estrutura do Wireframe


O wireframe foi desenvolvido no Figma e representa as telas principais da aplicação. Cada módulo possui uma interface com lista de itens à esquerda, detalhes à direita e botões de ação (Cadastrar / Excluir) na parte inferior.

Link:

<https://www.figma.com/design/lbB1fdzw4PM2joub9ZlzuC/WireFrame-AV2?node-id=681-248&t=gLM65SFGq9XoP295-1>

## Telas Principais

1. **Login** – Entrada no sistema com verificação de cargo. Conforme a figura abaixo



Wireframe da tela de Login. O título "Login" está centralizado no topo. Abaixo dele, há dois campos de entrada: "Usuário" e "Senha", cada um com um ícone de olho para alternar a visibilidade. No final, há um botão "Logar".

figura 1 - tela de login

2. **Seleção de Módulo** – Mostra os módulos disponíveis conforme permissão. Conforme a figura abaixo

## Selecionar o processo desejado

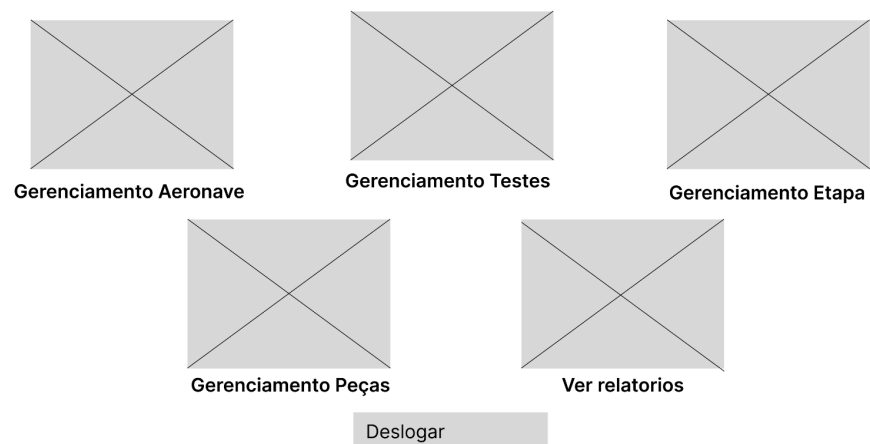


figura 2 - tela de seleção para engenheiros

3. **Aeronaves / Peças / Etapas / Testes / Funcionários / Relatórios** – Páginas de gerenciamento, cada uma com painel lateral e estrutura de listagem + detalhes. Conforme a figura abaixo:



figura 3 - Tela de gerenciamento de aeronave para Administradores

Essas telas seguem a mesma hierarquia e padrão visual, garantindo consistência e fácil aprendizado.

---

## Fluxo de Usuário (User Flow)

O User Flow foi criado no Miro, representando o caminho que o usuário percorre desde o login até o uso de cada módulo.

Link: [https://miro.com/app/board/uXjVJy\\_THc4=/?share\\_link\\_id=713838016922](https://miro.com/app/board/uXjVJy_THc4=/?share_link_id=713838016922)

### Estrutura do Fluxo

#### 1. Usuário acessa Login

- Insere usuário e senha.
- Sistema identifica o tipo de usuário. Representação visual

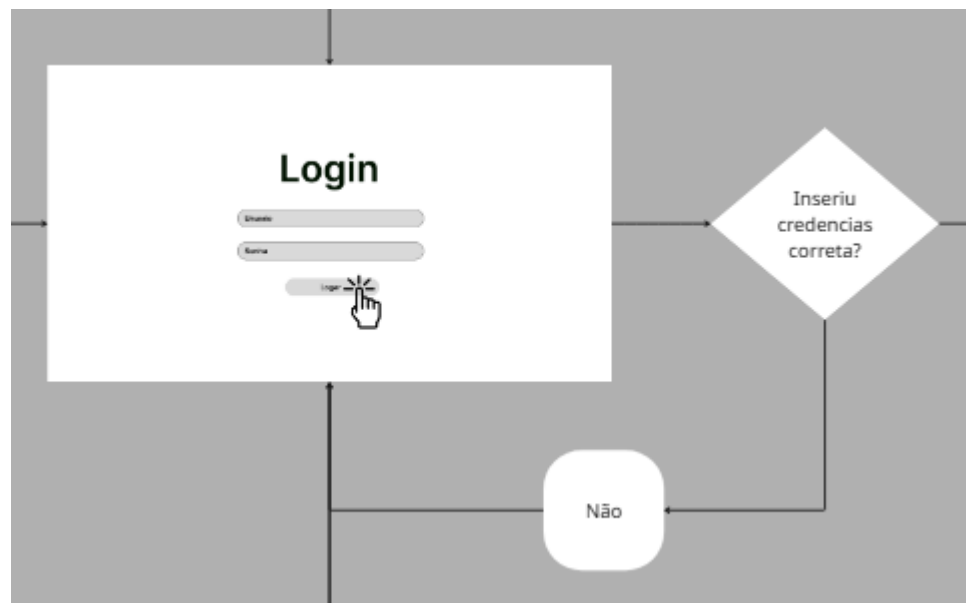


figura 4 - Fluxo do usuário ao fazer login

#### 2. Redirecionamento automático

- Para a tela de seleção com as opções disponíveis conforme o cargo.

- Representação visual:

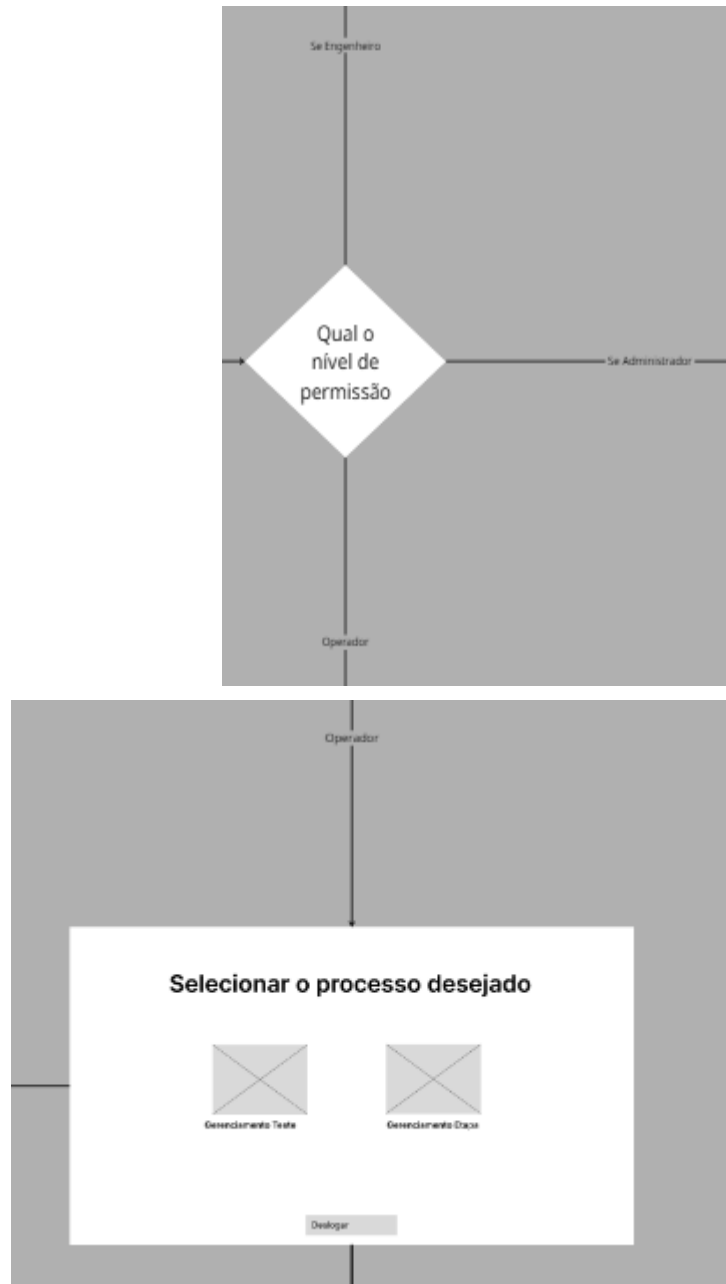


figura 5 e 6 - fluxo do para o cargo de operador após login

### 3. Acesso ao módulo

- Usuário escolhe uma das opções (ex: Etapas).
- Sidebar mostra apenas as seções liberadas.
- Representação visual:



figura 7 - Tela de gerenciamento de Teste

#### 4. Interação

- Cadastrar, visualizar ou excluir informações.
- Representação visual:

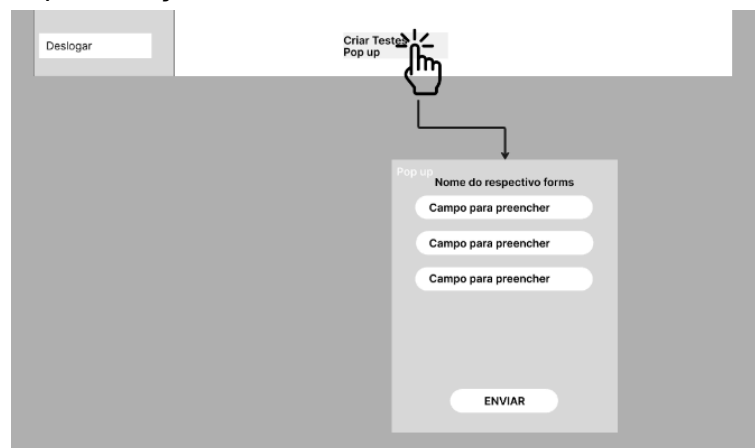


figura 8 - Pop de criação de teste

- Retornar à tela de seleção ou logout.

Esse fluxo visual representa claramente a lógica de permissões e navegação entre as telas.

# Desenvolvimento em React (SPA)

O desenvolvimento foi feito em React/Next.js, simulando uma SPA (Single Page Application).

A aplicação possui:

- Pastas individuais para cada módulo (/aeronave, /funcionario, /etapa, /peca, /teste, /relatorio).
- Um componente de barra lateral (LateralBar) dinâmico, que adapta as opções com base na role (Administrador, Engenheiro, Operador).
- Mock de dados para cada entidade (sem backend real).
- Uso de **useState** e **useSearchParams** para gerenciar estados e navegação.

Essa arquitetura permite rápida renderização, reuso de componentes e maior clareza no código.

---

## Hierarquia das Telas e Permissões

Cargo	Acesso	Funções
Administrador	Aeronaves, Peças, Etapas, Funcionários, Testes, Relatórios	Controle total
Engenheiro	Aeronaves, Peças, Etapas, Testes, Relatórios	Gestão técnica
Operador	Etapas, Testes	Execução e marcação de progresso

---



## Conclusão

O desenvolvimento do wireframe e do fluxo de navegação permitiu estabelecer uma base para a criação da GUI do sistema Aerocode. As decisões de design priorizaram usabilidade, modularidade e controle de acesso, tornando o sistema acessível e seguro para todos os perfis de usuário.

Com os usos do Wireframe e de um user flow bem construído, podemos ter uma base sólida para elaboração desse projeto e ter um melhor desenvolvimento e fluidez nos requisitos a se construir