

Desenvolvimento – TrakNor MVP

Documento atualizado em Jul/2025, focado apenas no desenvolvimento do MVP do TrakNor.

1. Estrutura de Repositório

Fluxo de Branches

- main – Versões estáveis e testadas.
- dev – Desenvolvimento contínuo.
- feature/ – Cada nova funcionalidade.
- bugfix/ – Correções de bugs.
- hotfix/ – Correções urgentes na main.

Estrutura de Pastas (MVP)

```
traknor/
├── backend/    # Django + DRF
│   ├── api/    # Viewsets / Serializers
│   ├── core/   # Domínio e casos de uso
│   ├── models/ # Modelos do banco
│   ├── tests/  # Pytest
│   ├── settings/ # Configurações Django
├── frontend/   # React + Vite
│   ├── components/ # Componentes reutilizáveis
│   ├── pages/     # Páginas
│   ├── api/       # Chamadas HTTP
│   ├── hooks/     # React Query
│   ├── assets/    # Estilos, ícones
├── docs/        # Documentação
├── docker-compose.yml
└── README.md
```

Removido do MVP: Celery, Redis, GraphQL e mobile.

2. Critérios de Aceitação (por módulo)

Login

- Autenticação via SessionAuth ou TokenAuth DRF
- Mensagens claras para erro de login
- Redirecionamento correto após login

Cadastro de Usuários

- Formulário funcional com campos obrigatórios
- Perfis: Admin, Gestor e Técnico
- Validação de e-mail duplicado

Cadastro de Equipamentos

- Campos obrigatórios validados
- Persistência correta e visualização listada
- Aviso para duplicidade

Criação de OS

- Formulário validado
- OS vinculada a equipamento e responsável
- Data de agendamento respeitada

Execução de OS

- Checklist obrigatório
- Status alterado automaticamente
- Arquivamento ao concluir

Dashboard

- KPIs renderizados com dados reais
- Filtros por status e data
- Layout responsivo apenas para desktop/tablet

Exportação de Relatórios

- PDF gerado corretamente
- Conteúdo formatado com cabeçalho do tenant

Permissões

- Acesso restrito de acordo com perfil

3. Testes e Qualidade

Tipos de teste:

- Unitários: funções principais
- Integração: Fluxos (Cadastro → Login → Dashboard)
- UI: Navegação, validações
- Exportação: PDF gerado com dados consistentes

4. Segurança

- HTTPS obrigatório
- Sem JWT nesta fase: apenas Token DRF e Session
- Rate limiting básico no servidor web

5. Performance (meta MVP)

- Dashboard: ≤ 1.5 s
- Listagem de equipamentos: ≤ 1 s (500 registros)
- Relatório PDF: ≤ 3 s

6. CI/CD

- GitHub Actions:
 - Lint + testes a cada push
 - Deploy manual no Lightsail/EC2
- Testes automatizados: cobertura mínima 70%

7. Responsividade

- Desktop ≥ 1024 px: Completo
- Tablet ≥ 768 px: Ajustes
- Mobile: Não contemplado; aviso exibido

8. Padrões de Qualidade (DoD)

- Possui testes unitários e de integração
- Documentação atualizada no README / Notion
- Revisão por pares aprovada
- Funciona em ambiente de staging
- Critérios de aceitação atendidos

Principais diferenças em relação ao documento antigo

- Remoção de JWT, Celery/Redis, aplicativo mobile, integrações IoT
- Enfoque em simplicidade para testar funcionalidades via código
- Workflow apenas com React + Django
- Deploy manual simplificado