

Recomendações de Melhores Práticas – Projeto Hubx

Resumo das melhorias prioritárias identificadas na estrutura atual do repositório:

Área	Problema Observado	Boas Práticas Recomendadas
Segurança & Configuração	<ul style="list-style-type: none">SECRET_KEY no Git.DEBUG=True.db.sqlite3 versionado.	<ul style="list-style-type: none">Variáveis em .env.DEBUG=False por padrão.Postgres em produção; ignorar SQLite.
Estrutura de Pastas	<ul style="list-style-type: none">venv/ versionado.Backend e Frontend misturados.	<ul style="list-style-type: none">Monorepo organizado: /backend e /frontend.Excluir venv/ do Git.
Dependências	<ul style="list-style-type: none">requirements.txt único sem pin de versões.	<ul style="list-style-type: none">Usar pip-tools ou pyproject.toml.Separar prod/dev/test.
Qualidade de Código	<ul style="list-style-type: none">Sem type hints, ruff/black, testes ou CI.	<ul style="list-style-type: none">Adicionar ruff/black/pre-commit.pytest com >90% coverage.GitHub Actions para lint/test.
Modelo de Domínio	<ul style="list-style-type: none">accounts.User inchado com campos redundantes.	<ul style="list-style-type: none">Aplicar DDD; mover campos extra para Perfil/Contato.
Internacionalização	<ul style="list-style-type: none">LANGUAGE_CODE='en-us'; TIME_ZONE='UTC'.	<ul style="list-style-type: none">Configurar para pt-BR e America/Sao_Paulo.
Transações	<ul style="list-style-type: none">ATOMIC_REQUESTS não configurado.	<ul style="list-style-type: none">Definir ATOMIC_REQUESTS=True no database.
Persistência de Mídia	<ul style="list-style-type: none">media/ contém arquivos versionados; .gitignore incompleto.	<ul style="list-style-type: none">Usar storage externo (S3/GCS); usar django-storages.
Documentação	<ul style="list-style-type: none">README raso; sem docs de arquitetura.	<ul style="list-style-type: none">Expandir README; gerar OpenAPI com DRF; adicionar testes de integração.

Próximos Passos Rápidos

- Higienizar repositório: remover artefatos locais (venv, db.sqlite3, media) e reforçar .gitignore.
- Configurar settings via django-environ e criar arquivo .env.example.
- Implantar pipeline de CI (GitHub Actions) com lint, testes e build de container.
- Separar backend e frontend em pastas dedicadas.
- Iniciar migração para Tailwind + HTMX seguindo wireframes.