

Seleção de Desenvolvedores Teste Técnico

HubLocal

https://hublocal.com.br

1. Proposta

Implementação de um sistema que deve:

- a. Logar/Criar usuários
- b. Criar/Listar/Deletar/Editar empresas (usuário logado)
- c. Criar/Listar/Deletar/Editar locais pertencentes a uma empresa (usuário logado)

2. Estimativa de Entrega

01/05/2025

3. Tecnologias/Ambiente

 a. React JS ou Next JS -> no Front-End com as seguintes techs como diferencial.

Obs: as techs listadas abaixo fazem parte de nossos projetos diariamente. Nada impede que você utilize outras como **Chakra UI e Context API em vez de um Material UI e Redux**, e assim por diante.

- i. Material UI https://mui.com/material-ui/
- ii. Styled Component https://styled-components.com/
- iii. Redux (Ou Zustand) https://redux.js.org/
- iv. **Opcional:** Redux Persist (Caso use o Zustand, ignorar) https://github.com/rt2zz/redux-persist
- v. Axios https://axios-http.com/ptbr/docs/intro
- b. Nest JS no Back-end com TypeORM ou Prisma e Postgres. As que utilizamos e desejamos como diferencial. No lugar de TypeORM uma boa pedida seria Prisma. Criar migrations e deixá-las no teste.
- c. Escolher algum provedor de hospedagem a seu critério. Caso não consiga, deixe um .env.example com as envs utilizadas no projeto.
- d. Deixar instruções detalhadas no README de cada projeto

4. Back-End

a. Entidades:

- i. Usuários:
 - Colunas: Nome, Email e Senha.
 - Relacionamentos: One To Many com Empresas.

ii. Empresas:

- Colunas: Nome, Website e CNPJ.
- Relacionamentos: Many To One com Usuários
 e One To Many com Locais.

iii. Locais:

- Colunas: Nome, CEP, Rua, Número, Bairro,
 Cidade e Estado.
- Relacionamentos: Many To One com Empresas.

b. Auth:

- i. Validar todas as rotas dos controllers com JWT;
- ii. Ao usuário logar, o token deve ser retornado para ser guardado para as próximas requests no front-end.

5. Front-End

a. Validações

- i. campos dos formulários, principalmente que envolvem números;
- ii. Adicionar máscaras nos campos que precisem, no caso sendo um deles o **CNPJ**.

b. Controle de estados e persistência

- i. Usar Redux, Context API ou qualquer outro framework;
- ii. Usar Redux Persist ou outro meio para persistir dados de estado.

c. Usar Lib de Feedback

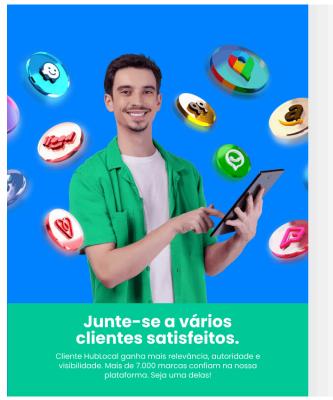
i. notistack, react-toastify.

6. Telas

Link do Figma:

https://www.figma.com/file/aEgfeSSNgaycj2533zay6j/my-companies-test e-hublocal?node-id=18%3A1128&t=MMokX7s0Cm6jYLUx-1

1. Tela de Login

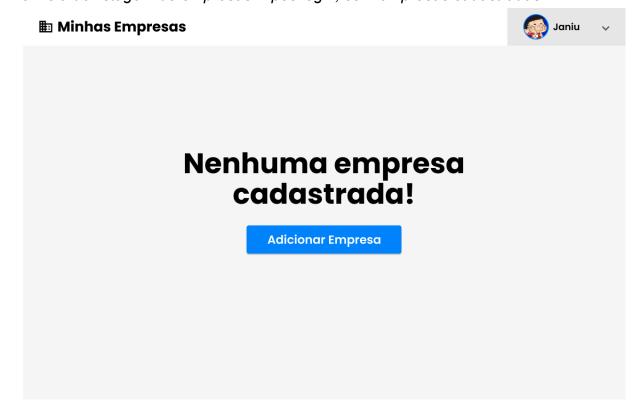


Q HubLocal
Email
Senha
LOGAR
CRIAR CONTA

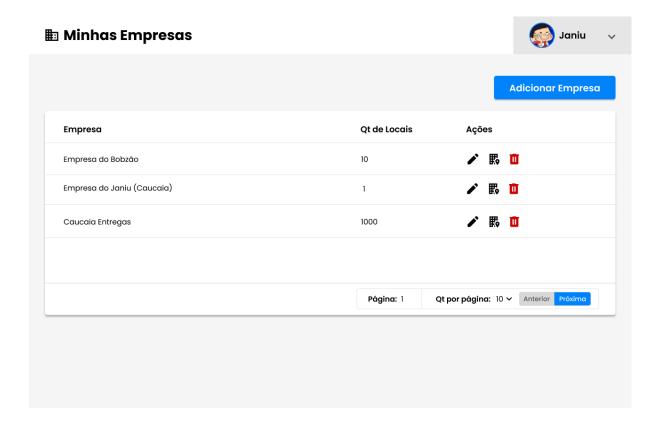
2. Tela de Registro



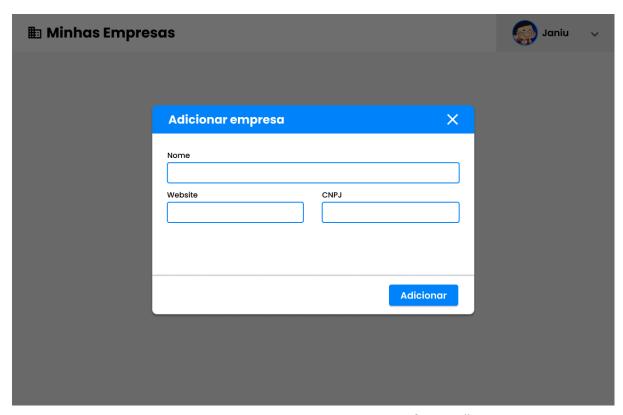
3. Tela de listagem de empresas. Após login, sem empresas cadastradas



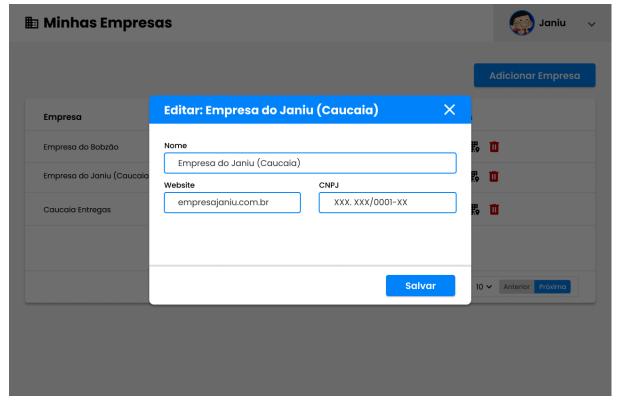
4. Tela de listagem de empresas



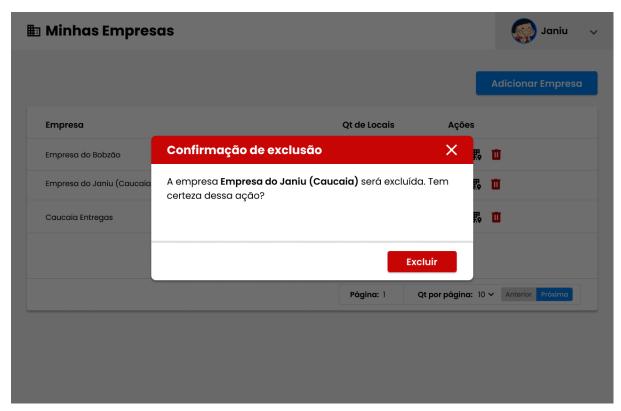
5. Tela de listagem de empresas, com modal de adicionar empresa



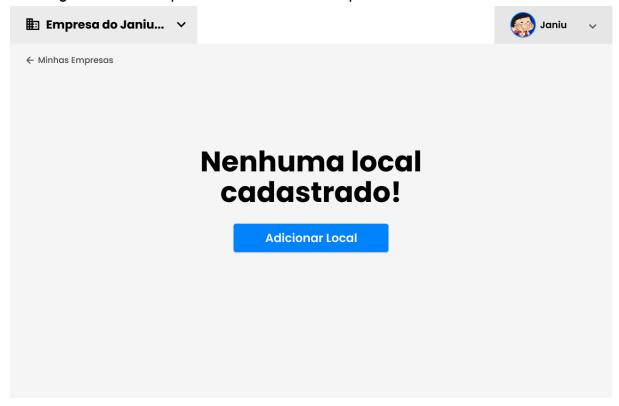
6. Tela de listagem de empresas, com modal de editar informação de uma empresa



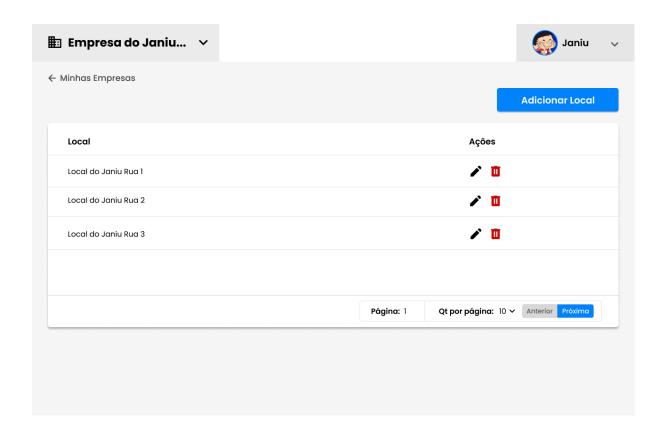
7. Tela de listagem de empresas, com modal de confirmação de exclusão de empresa



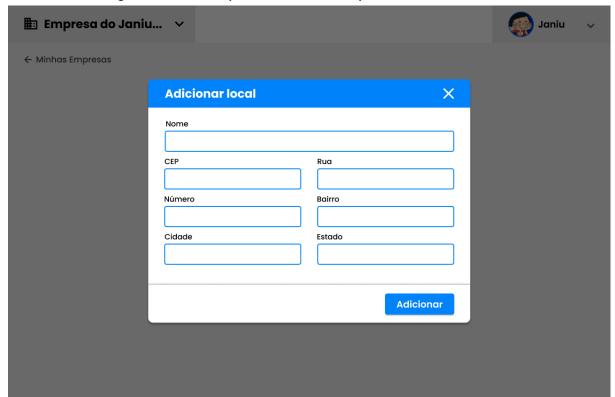
8. Listagem de locais após selecionado uma empresa



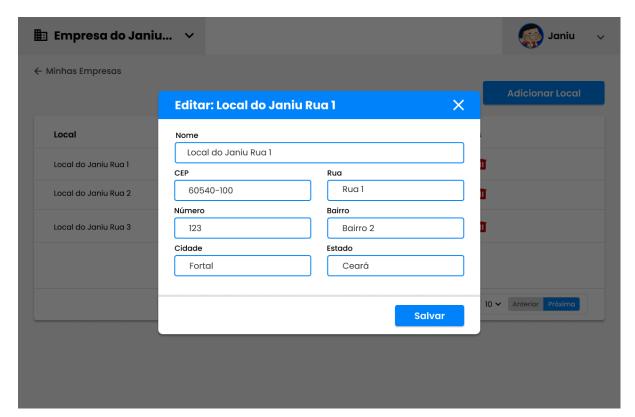
9. Tela de listagem de locais após selecionar empresa



10. Tela de listagem de locais após selecionar empresa, modal de adicionar local



11. Tela de listagem de locais após selecionar empresa, modal de editar local



12. Tela de listagem de locais após selecionar empresa, modal de exclusão de empresa

