



UNIVERSIDADE FEDERAL DO CEARÁ

CAMPUS QUIXADÁ

QXD0020 - DESENVOLVIMENTO DE SOFTWARE PARA WEB

PROF. FRANCISCO VICTOR DA SILVA PINHEIRO

**TRABALHO FINAL
ENTREGA 2 – FRONTEND**

1. INTEGRANTES

Matrícula	Nome	E-mail
516384	Luiz Eduardo Paiva Ribeiro	eduardopaiva11@alu.ufc.br

2. TÍTULO DO PROJETO

Tem vaga aí?

3. ESTRUTURA DO PROJETO

- Projeto criado com **React + Vite + TypeScript**
- Configuração inicial concluída
- Principais pastas: src/presentation (UI, páginas, context), src/application (use cases), src/domain (entities / repositories), src/infrastructure (repositórios e cliente HTTP).
Dependências principais: React, react-router, Tailwind, TypeScript, Vite.
Comandos para rodar localmente:
- npm install
- npm run dev (abre http://localhost:5173 por padrão)
Variáveis de ambiente:
- VITE_API_URL — URL do backend (default: http://localhost:4000/api) — ver src/infrastructure/api/HttpClient.ts.

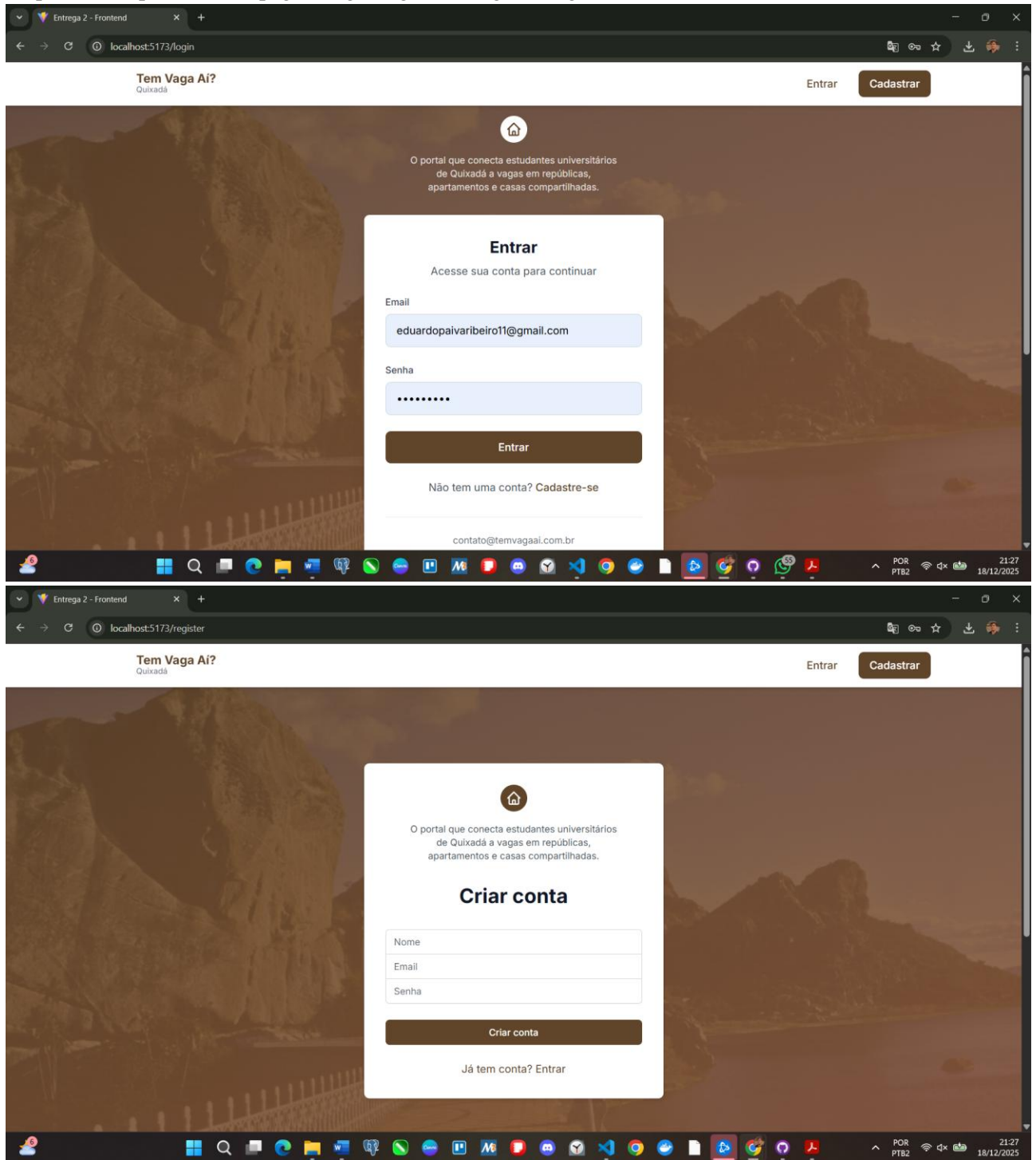
4. ROTAS IMPLEMENTADAS

- / — Home pública (listagem de anúncios)
- /login — Tela de Login
- /register — Tela de Registro
- /ad/:id — Tela de detalhe do anúncio
- /home-logado — Home após login (feed personalizado)
- /meus-anuncios — Área do usuário: listar seus anúncios
- /favoritos — Lista de anúncios favoritos
- /novo-anuncio — Criar novo anúncio
- /editar-anuncio/:id — Editar anúncio existente
- /perfil-info — Meu perfil — informações
- /perfil-seguranca — Meu perfil — segurança (alterar senha)

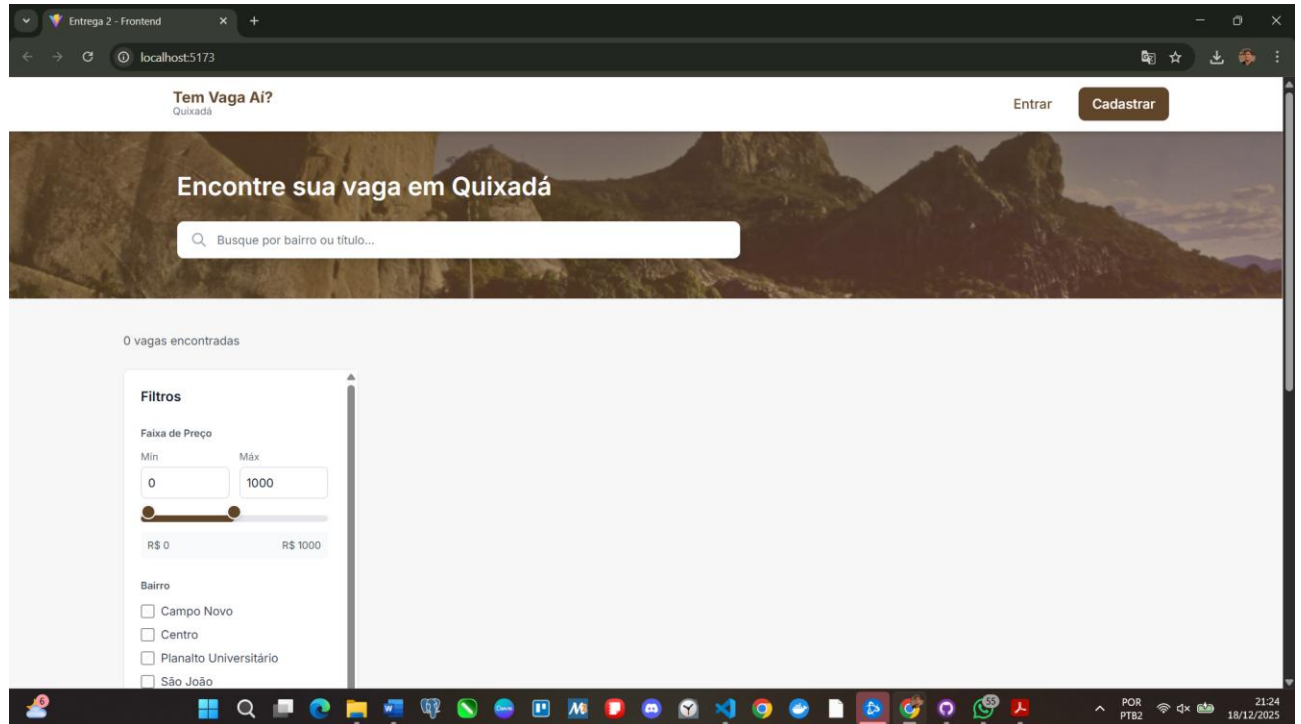
5. TELAS ESTÁTICAS OBRIGATÓRIAS (PRINTS)

- Tela de Login/Registro

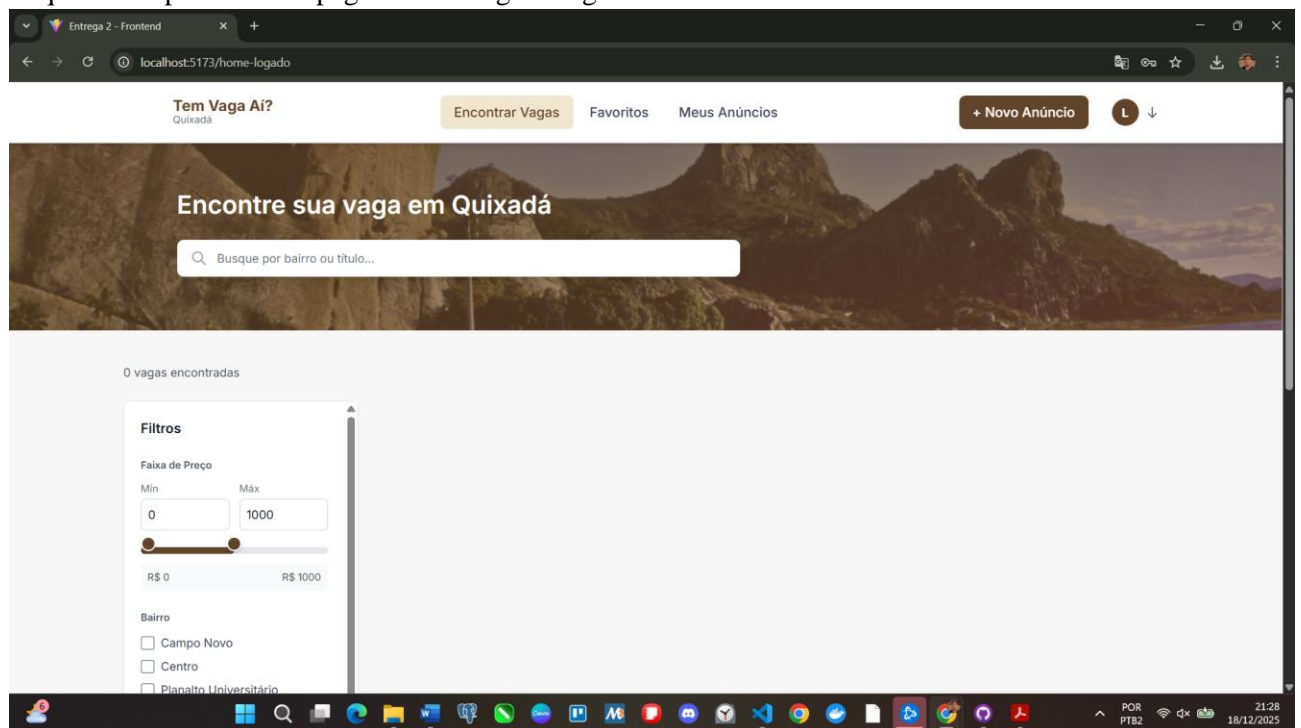
Arquivos: src/presentation/pages/LoginPage.tsx, RegisterPage.tsx



- Tela de Feed/Listagem (Home pública)
Arquivo: src/presentation/pages/HomePage.tsx



Arquivo: src/presentation/pages/HomeLogadoPage.tsx



- Tela de Detalhe do Anúncio (/ad/:id)
Arquivo: src/presentation/pages/AdDetailsPage.tsx

The image displays two screenshots of a web application interface for creating a new advertisement. The browser address bar shows the URL `localhost:5173/novo-anuncio`.

Top Screenshot: "Novo anúncio" form

The form is titled "Publicar moradia" and "Novo anúncio". It contains the following fields:

- Título:** Ex.: Suite mobiliada perto do campus
- Categoria:** Venda
- Vendedor/Anunciante:** Seu nome ou república
- Localização:** Bairro, cidade
- CEP (opcional):** Ex.: 63000-000
- Descrição:** Destaque comodidades, regras e proximidades importantes
- Preço (opcional):** Ex.: 750
- Quartos (opcional):** Ex.: 2

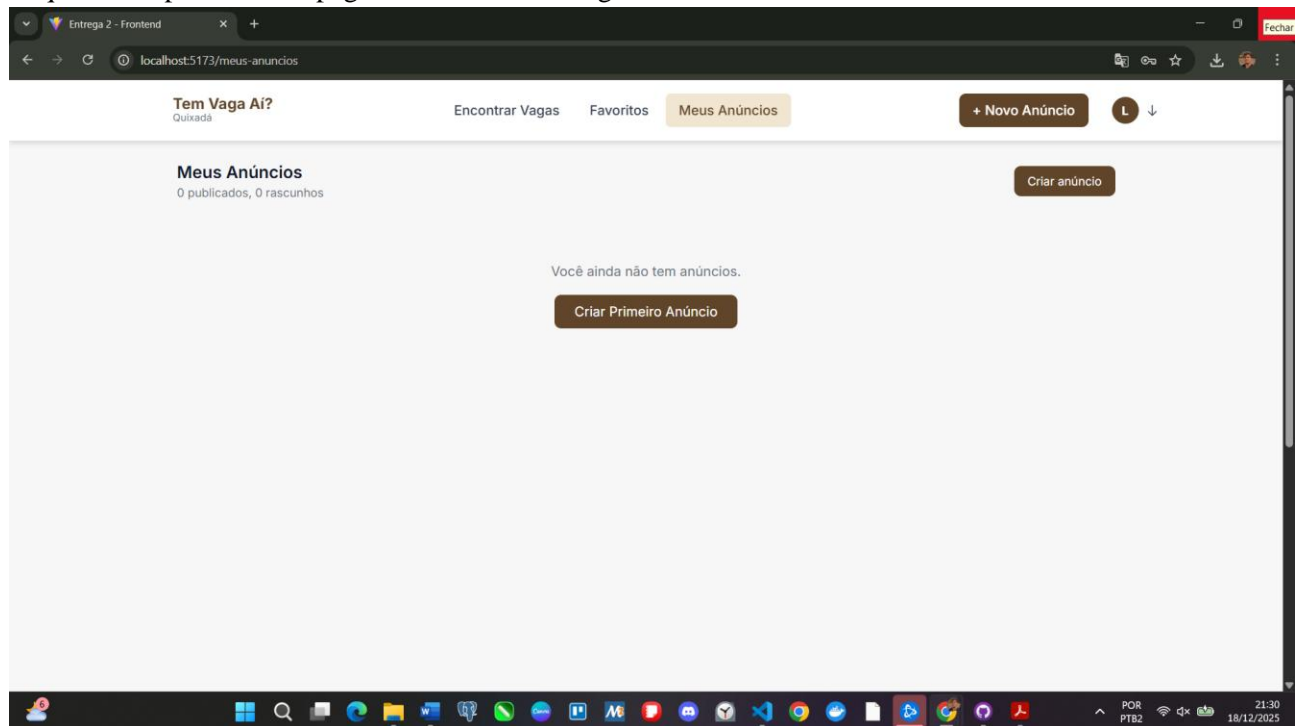
Bottom Screenshot: Continuation of the form

This section continues the form with the following fields:

- Banheiros (opcional):** Ex.: 1
- Regras do Imóvel:**
 - ☐ Proibido fumar
 - ☐ Proibido animais
 - ☐ Proibido festas
 - ☐ Silêncio após 22h
 - ☐ Visitas permitidas
- Outras regras (ex.: Entrada após 18h, Limpeza compartilhada...):**
- Comodidades e Inclusões:**
 - ☐ Água inclusa
 - ☐ Luz inclusa
 - ☐ Internet inclusa
 - ☐ Mobiliado
 - ☐ Ar-condicionado
 - ☐ Lavanderia
- Outras comodidades (ex.: Academia, Piscina, Churrasqueira...):**
- Fotos do imóvel:** 0/20
 - Clique ou arraste para enviar
 - Até 20 fotos em JPG ou PNG

At the bottom of the form, there are three buttons: "Publicar anúncio", "Salvar rascunho", and "Cancelar".

- Tela de Dashboard / Meus Anúncios (/meus-anuncios)
Arquivo: src/presentation/pages/MeusAnunciosPage.tsx



6. FORMULÁRIOS COM VALIDAÇÃO

Login

- Campos: email, password.
- Validação: checagem de preenchimento (front-end) antes de chamar login. Exibe erro local ou do contexto.

Registro

- Campos: name, email, password.
- Validação: checagem de preenchimento (front-end) antes de chamar register.

Novo Anúncio / Editar Anúncio

- Campos principais: title, description, price, location, cep, bedrooms, bathrooms, type (categoria), rules, amenities, imagens (upload).
- Validações implementadas no front-end:
 - Campos obrigatórios verificados (ex.: título e descrição antes de publicar).
 - price, bedrooms, bathrooms devem ser enviados como valores numéricos (conversão/manuseio no frontend).
 - **Limite de imagens:** máximo de **20** imagens — caso exceda, o usuário recebe mensagem de erro local.
 - Possibilidade de salvar rascunho / publicar.
- Upload de imagens: conversão para Base64 no front-end (FileReader) e envio ao backend via httpClient.

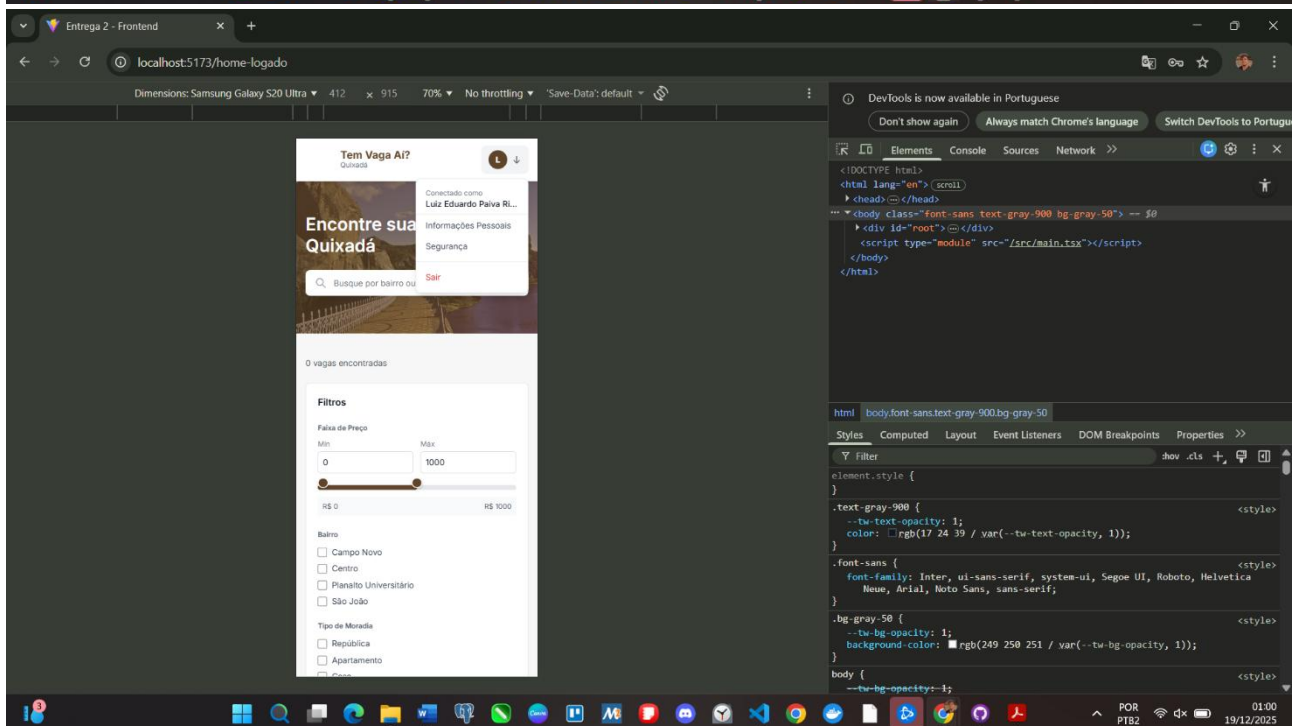
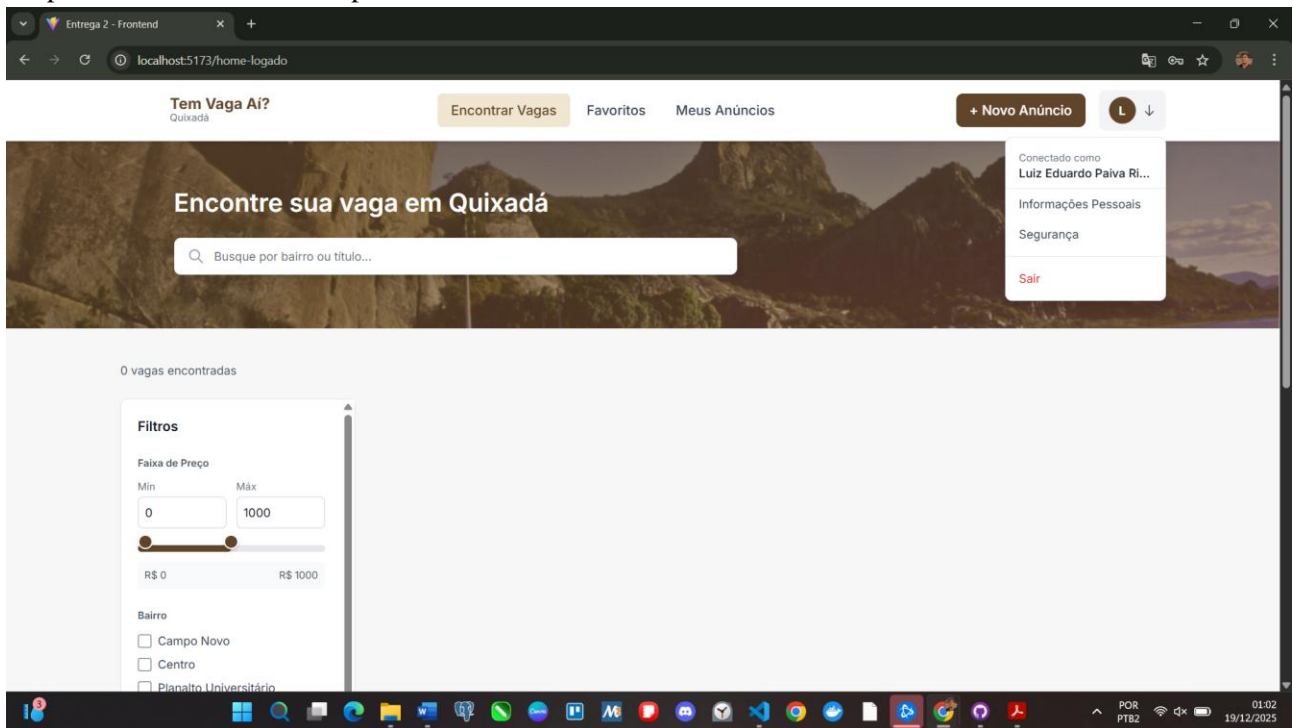
Perfil / Segurança

- Troca de senha: verificação se `newPassword === confirmPassword` no front-end antes de submeter.

7. LAYOUT RESPONSIVO

O projeto usa Tailwind CSS para estilização e responsividade (classes utilitárias, breakpoints padrão).

Componentes e containers usam `max-w-*`, `mx-auto`, `flex`, `grid`, e utilitários responsivos (`sm:`, `md:`, `lg:`) para adaptar entre mobile e desktop.



8. SERVICES MOCKADOS

- O frontend consome uma API via HttpClient (src/infrastructure/api/HttpClient.ts) apontando para VITE_API_URL (padrão http://localhost:4000/api).
- Em vários pontos há fallback/local persistence: o AuthContext e FavoritesContext persistem informações em localStorage (temVagaAi.user, temVagaAi.token, temVagaAi.favorites) quando o backend não responde — permitindo comportamento offline/sem backend imediato.
- Repositório de anúncios (src/infrastructure/repositories/AdsRepository.ts) realiza chamadas HTTP (/ads, /users/me/ads, etc.). Para demonstração sem backend, o app recorre ao localStorage em algumas partes.
- Conclusão para o template: dados via API HTTP; fallback para localStorage quando necessário.

9. COMPONENTES REUTILIZÁVEIS

- Navbar — navegação global (src/presentation/components/Navbar.tsx)
- Footer — rodapé (src/presentation/components/Footer.tsx)
- AdGallery — galeria de imagens do anúncio (AdGallery.tsx)
- AdHeader — cabeçalho do anúncio (preço, local) (AdHeader.tsx)
- AmenitiesBlock — bloco de comodidades (AmenitiesBlock.tsx)
- RulesBlock — bloco de regras do imóvel (RulesBlock.tsx)
- DescriptionBlock — descrição formatada (DescriptionBlock.tsx)
- ContactModal / ContactPanel — interface para contato entre interessado e locador (ContactModal.tsx, ContactPanel.tsx)
- AuthModal — modal de autenticação (quando aplicável)

(Estes componentes são usados nas páginas para composição e reutilização).

11. REPOSITÓRIOS

- Frontend: <https://github.com/eduardoribeir/web-2025-2-tem-vaga-ai-ribeiro-frontend/tree/main>
- Backend <https://github.com/eduardoribeir/web-2025-2-tem-vaga-ai-ribeiro-backend/tree/main>