## VetCare - Frontend (Vite + React + TypeScript)

Proyecto full‑stack de una aplicación de veterinaria. En esta etapa se implementó el frontend completo de la landing y las pantallas de autenticación (Login y Registro) con una estética cálida y moderna, coherente con la marca.

Tecnologías

- React 19 + TypeScript + Vite

- Enrutamiento con `react-router-dom`

- PostCSS + `@tailwindcss/postcss` (Tailwind v4 habilitado; se usa CSS utilitario propio y variables de color)

Características actuales

- Landing page fiel al diseño de referencia:

  - `Navbar` con logo configurable y botón “Iniciar Sesión”.

  - `Hero` con imagen destacada de la mascota.

  - Sección de beneficios con íconos reemplazables.

  - Grid de servicios en tarjetas.

  - Sección de llamada a la acción (CTA) con gradiente.

- Autenticación (demo, sin backend por ahora):

  - Pantalla de Login con diseño futurista/glassmorphism, inputs controlados y navegación.

  - Pantalla de Registro con campos: nombre, email, teléfono y contraseña; inputs controlados y navegación.

  - Rutas disponibles: `/` (Landing), `/login`, `/register`.

- Paleta de color cálida (naranja/terracota/dorado) con acentos verde esmeralda, degradados suaves y fondos orgánicos con line‑art de animales.

Requisitos

- Node.js 18+ (recomendado 20+)

Estructura relevante

frontend/

  public/

    images/              # Activos reemplazables (logo, íconos, hero)

  src/

    components/          # Navbar, Hero, Benefits, Services, CTA

    pages/               # Landing, Login, Register

    main.tsx             # Rutas y montaje de la app

    index.css            # Estilos base, variables y utilidades

Decisiones de diseño

- Estética minimalista y sofisticada con glassmorphism en tarjetas.

- Degradados cálidos y acentos verdes suaves; sombras sutiles y borde luminoso.

- Capas decorativas con `pointer-events: none` para no bloquear la interacción.

Estado actual (sin backend todavía)

- Los formularios de Login y Registro son interactivos (inputs controlados). Al enviar muestran un flujo de demo y navegan.

- Falta conectar con el backend (Express + MySQL) para autenticación real, validaciones del lado servidor y persistencia.

Próximos pasos

- Implementar backend en Express + MySQL (registro, login, sesiones/JWT).

- Validación y mensajes de error reales.

- Proteger rutas privadas y estado global de autenticación.

- Integración de llamadas API desde el frontend.