Estructura del proyecto:

## **Lenguajes de programación**

* **Frontend:** **TypeScript**
* **Backend:** **Python** (usado en FastAPI).

## **🔹 Frameworks y librerías principales**

* **Frontend:**  
  + **React** → framework/biblioteca para construir interfaces web.
  + **Vite** → *build tool* y servidor de desarrollo para React.
  + **Tailwind CSS** → framework de estilos CSS utilitario.
* **Backend:**  
  + **FastAPI** → framework moderno en Python para construir APIs rápidas y tipadas.

## **🚀 Justificación tecnológica**

* **React + Vite + TypeScript**  
   Usamos **React** porque es el estándar actual para interfaces web modulares y reactivas; **Vite** porque acelera el desarrollo con compilación muy rápida; y **TypeScript** porque nos da tipado estático, evitando errores en un sistema con carrito, productos y chat.
* **Tailwind CSS**  
   Lo elegimos porque nos permite crear una interfaz moderna y responsiva en poco tiempo, sin sacrificar consistencia de diseño. Es más ágil que escribir CSS desde cero.
* **FastAPI (backend)**  
   Elegimos **FastAPI** porque es un framework rápido, tipado y moderno en Python. Genera documentación automática y facilita conectar el chatbot con modelos de IA, que es un objetivo clave de nuestro proyecto.

Elegimos Vite porque es el más rápido, moderno y sencillo de configurar. Nos permite enfocarnos en la lógica de la tienda y el chatbot

# **Qué es el chatbot en este proyecto**

* Es un **widget React** embebido
* Soporta **modo real** (con backend) y **modo mock** (sin backend) para poder desarrollar y presentar siempre.

\**Se conectará en un futuro con OpenAi\* Lei que es lo mejor*

**ESTRUCTURA DE CARPETAS**  
lib/

**types.ts** # Tipos compartidos (Product, Category, CheckoutItem, ChatMessage)

**api.ts** # Cliente HTTP + fallback a mock

**mock**/

**products.ts** # Datos de ejemplo para funcionar sin backend /

Permite seguir desarrollando/presentando sin backend.

**components**/

**Navbar.tsx** # Menú y filtros por categoría + botón abrir carrito

**ProductCard.tsx** # Tarjeta de producto (imagen, precio CLP, botón Agregar)

**CartDrawer.tsx** # Carrito lateral (cantidad, total, pagar)

**ChatWidget.tsx** # Bot flotante 💬 con historial y envío

**Footer.tsx**  # Enlaces básicos

**pages/**

**Home.tsx**  # Hero + grilla filtrada de productos

**App.tsx**  # Orquestación: estado de carrito/categoría y handlers

**main.tsx** # Bootstrap React

**index.css** # Tailwind directives

Añadir **.env si no está**

VITE\_USE\_MOCK=1 # 1 = usa datos locales y chat simulado

VITE\_API\_BASE=http://127.0.0.1:8000/api # usar si hay backend