Got it ✅ I’ll generate a **structured, polished, final prompt** you can paste directly into Cursor. It will clearly define context, inputs, user flow, system behavior, and output requirements — no fluff, just complete, actionable steps.

Here’s the **final Cursor prompt** for your vaccine information application:

**Prompt for Cursor:**

You are building a **Vaccine Information Web Application** that can later be extended into a mobile app. Follow the specifications below carefully.

**Context**

* Purpose: Provide accurate, FDA-sourced vaccine information to users in a clean, easy-to-navigate interface.
* Audience: General public, parents, caregivers, and healthcare professionals seeking vaccine details.
* Platform: Start as a **web application** (responsive, modern UI), designed so it can be later packaged as a **mobile app** (using React Native or similar).

**Accepted Inputs**

* User selects a vaccine from a **dropdown list** of FDA-approved vaccines.
* Data must be fetched in real-time from FDA’s site (starting point: <https://www.fda.gov/vaccines-blood-biologics/vaccines/vaccines-licensed-use-united-states>).
* Explore if FDA offers an API or structured dataset. If not, scrape or parse FDA.gov pages.
* Inputs: Dropdown selection only (no free-text input required).

**User Flow (Step-by-Step)**

1. User lands on the home page with a brief description of the app’s purpose.
2. User sees a dropdown list of FDA-approved vaccines.
3. User selects a vaccine.
4. App queries FDA (and/or other trusted data sources, e.g., clinical trial databases).
5. App displays a results page with:
   * Date vaccine was added to childhood schedule (if applicable)
   * Manufacturer
   * Clinical trial period (duration of monitoring)
   * Number of participants in the trial
   * Age range of participants
   * Adverse effects discovered + rate of occurrence
6. User can return to the dropdown to query another vaccine.

**System Behavior**

* **Frontend**:
  + Use **React (with TypeScript)** for a scalable, responsive UI.
  + Dropdown component populated dynamically from FDA data.
  + Results displayed in a structured card/table format.
* **Backend**:
  + Use **Node.js + Express** for API requests and data processing.
  + Create an API layer that queries FDA.gov and parses vaccine details.
  + Store frequently accessed vaccine data in a database for caching and faster queries.
* **Database**:
  + Use **PostgreSQL** for structured data (scalable, reliable).
  + Include tables for vaccines, manufacturers, clinical trial data, and adverse effects.
* **Mobile-readiness (Phase 2)**:
  + Codebase should be structured so frontend logic can be ported to **React Native**.
  + API/backend should remain the same for both web and mobile clients.
* **Error Handling**:
  + Show clear error messages if FDA data cannot be fetched.
  + Provide fallback information if live API is unavailable (cached DB data).

**Output Requirements**

* Deliver a **scaffolded full-stack application** with:
  + React + TypeScript frontend
  + Node.js/Express backend
  + PostgreSQL database setup
* Include a clean, professional UI with cards/tables for results.
* Code should be modular, well-documented, and production-ready.
* Follow modern best practices (REST API, environment variables for API keys, etc.).
* Provide clear **README documentation** for setup and deployment.

**Build this application according to the above specifications.**