WhatsApp Product Review Collector

Preferred Tech Stack: Python (FastAPI/Flask/Django), React, Postgres

Overview

Build a small full-stack application where users submit product reviews over WhatsApp. Your backend must receive these messages, process the conversation, store reviews in a Postgres database, and expose an API that the React frontend will consume to display the reviews.

You are free to choose any Python web framework (FastAPI preferred for ease).
For WhatsApp integration, you can use **Twilio WhatsApp Sandbox** (free and quick to set up).

What you need to build

1. WhatsApp -> Server Flow

- A user messages a WhatsApp number (Twilio sandbox).
- Twilio forwards the message to your backend webhook.
- Your backend must run a simple conversation flow to collect:
    1. **Product name**
    2. **User name**
    3. **Product review**

Conversation Example -

```
User: Hi
Server: Which product is this review for?

User: iPhone 15
```

```
Server: What's your name?

User: Aditi
Server: Please send your review for iPhone 15.

User: Amazing battery life, very satisfied.
Server: Thanks Aditi -- your review for iPhone 15 has been recorded.
```

2. Store Review in Postgres

Your Postgres table must contain:

Column	Type
Id	unique id
contact_numbr	text
user_name	text
product_name	text
product_review	text
created_at	timestamp

3. Backend APIs

Your backend must expose:

GET /api/reviews
Returns all stored reviews in JSON.

```
[
  {
    "id": 1,
    "contact_number": "+1415XXXXXXX",
    "user_name": "Aditi",
    "product_name": "iPhone 15",
    "product_review": "Amazing battery life",
    "created_at": "2025-11-17T12:34:56Z"
  }
]
```

4. React Frontend

Build a simple UI that:

- Fetches reviews from `/api/reviews`
- Displays them in a clean list/table:
    - User name
    - Product
    - Review
    - Timestamp

Minimal styling is fine.

5. Expected Deliverables

- Submit a GitHub repo (or two repos)
- Screencast of the working prototype
- README.md in project to explain to how run the backend

6. Time Expectation

This assignment is scoped so it can be completed comfortably in **~2 days**.