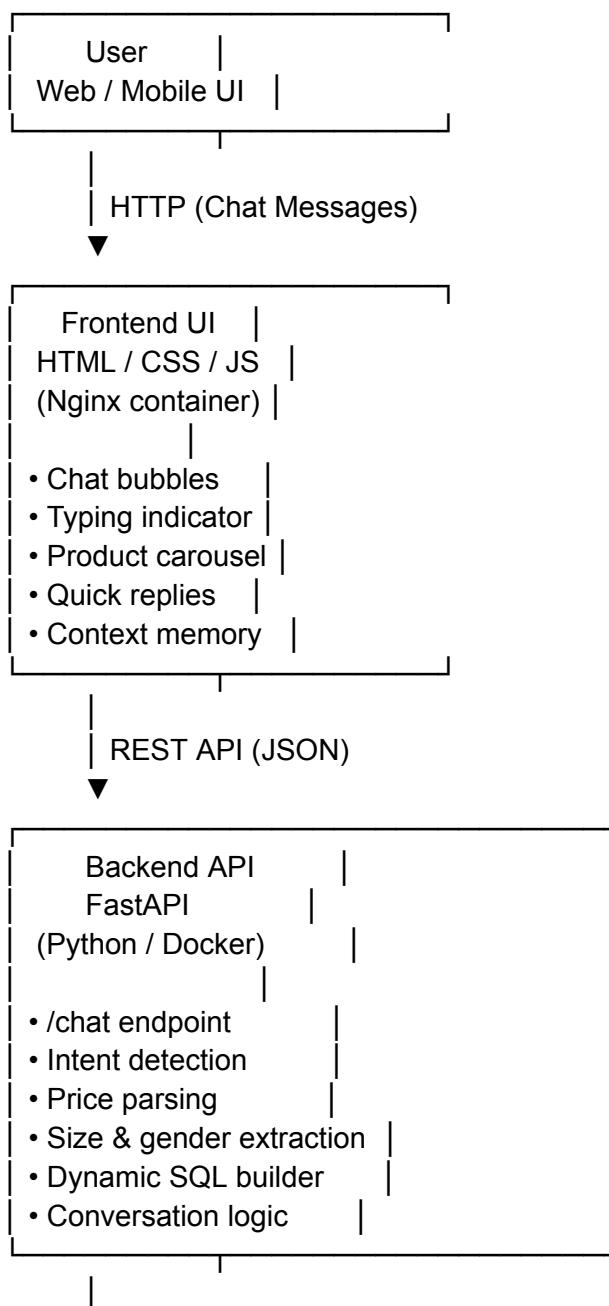
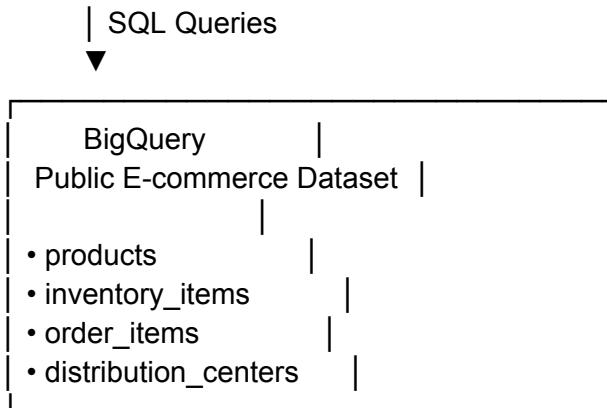


Architecture Overview — E-commerce Chatbot

1. High-Level System Architecture





2. Frontend Responsibilities

Technology

- HTML
- CSS (mobile-first, chat-style UI)
- Vanilla JavaScript
- Served via Nginx (Docker)

Key Responsibilities

- Render chat bubbles (user & bot)
- Show typing indicator ("Bot is typing...")
- Display products as horizontal carousels
- Render quick reply buttons
- Maintain lightweight conversation context
- Send user messages to backend API

3. Backend Responsibilities

Technology

- FastAPI (Python)
- Google BigQuery client
- Dockerized service

Key Responsibilities

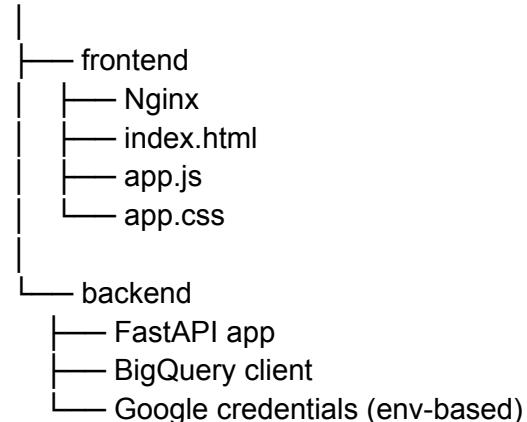
- Natural language parsing (rules-based)
- Intent detection:
 - Search
 - Gift recommendation
 - Comparison
 - Price / size filtering
- Constraint extraction:
 - Price (under / over / exact)
 - Gender → department (Men / Women)
 - Size (S, M, L, XL, XXL)
 - Category keywords (jackets, shirts, etc.)
- Dynamic SQL query construction
- Graceful “no results” handling
- Structured JSON responses:
 - reply
 - products
 - Quick_replies

4. Data Flow (Request → Response)

1. User types a message
2. Frontend sends request to /chat
3. Backend parses intent & constraints
4. SQL query generated dynamically
5. BigQuery executes query
6. Results returned to backend
7. Backend formats response
8. Frontend renders reply + products

5. Deployment Architecture

Docker Compose



6. Scalability & Production Notes

- **Stateless backend** → horizontally scalable
- **BigQuery** used for analytics & discovery
- **Transactional DB** (Postgres) recommended for:
 - Orders
 - Users
 - Carts

- **Redis** recommended for:
 - Conversation context
 - Query caching
- **LLM integration** can replace rule-based parsing
- **Monitoring** via:
 - Logs
 - Request metrics
 - Error tracking

7. Design Philosophy

- Chat-first UX reduces cognitive load
- Guided discovery supports vague queries
- Carousels preserve conversational flow
- Quick replies accelerate common actions
- Mobile-style UI mirrors familiar messaging apps

Notes

- Product images and stock are mocked due to dataset limitations
- Checkout flow is conversational (no payment integration)
- Designed for clarity, explainability, and extensibility