

Building the Ultimate Product Recommendation Agent

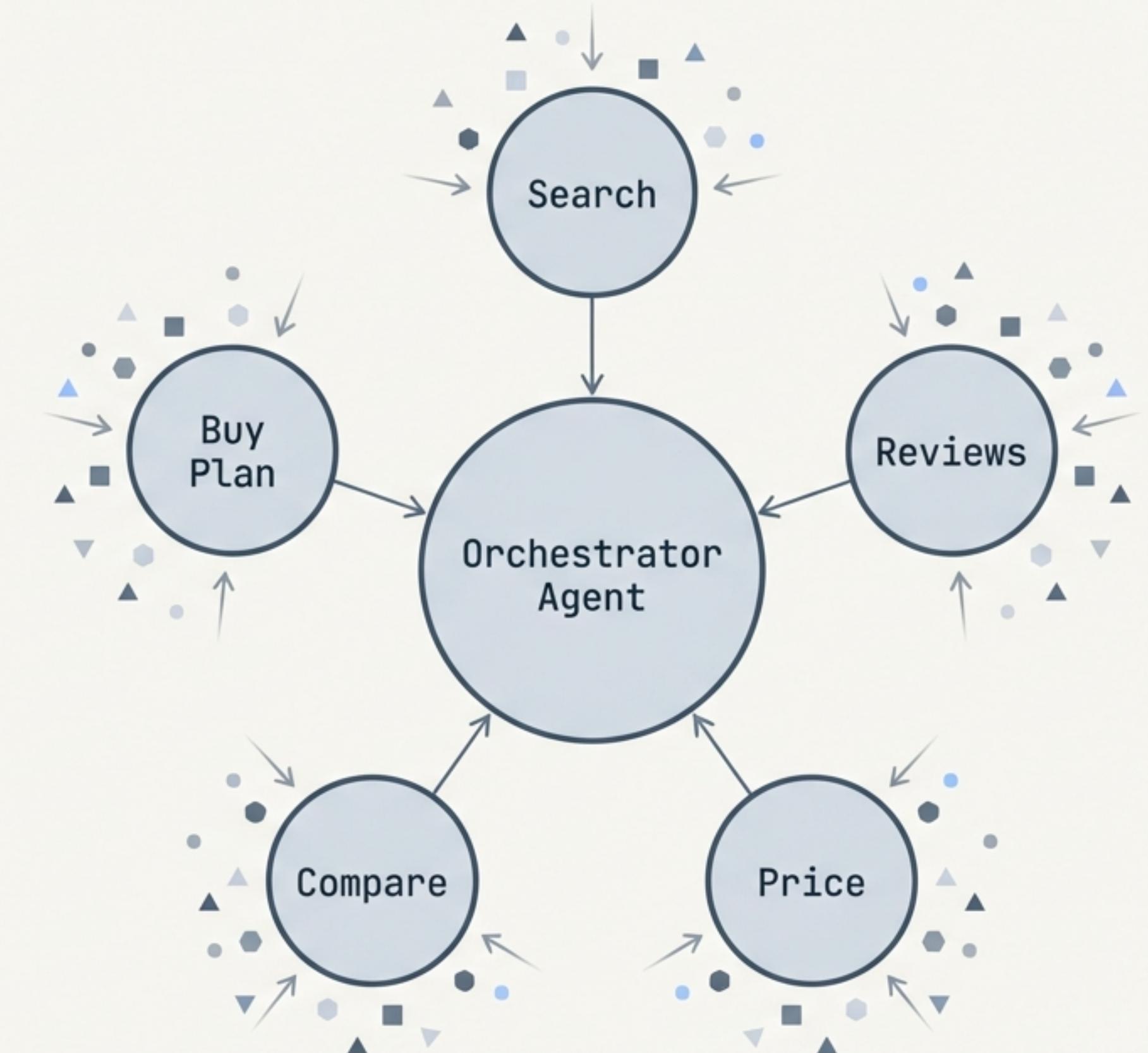
A Master Implementation Guide
using Google ADK & Gemini 1.5 Pro

FRAMEWORK: Google ADK v0.1.0

CORE AI: Google Gemini 1.5 Pro

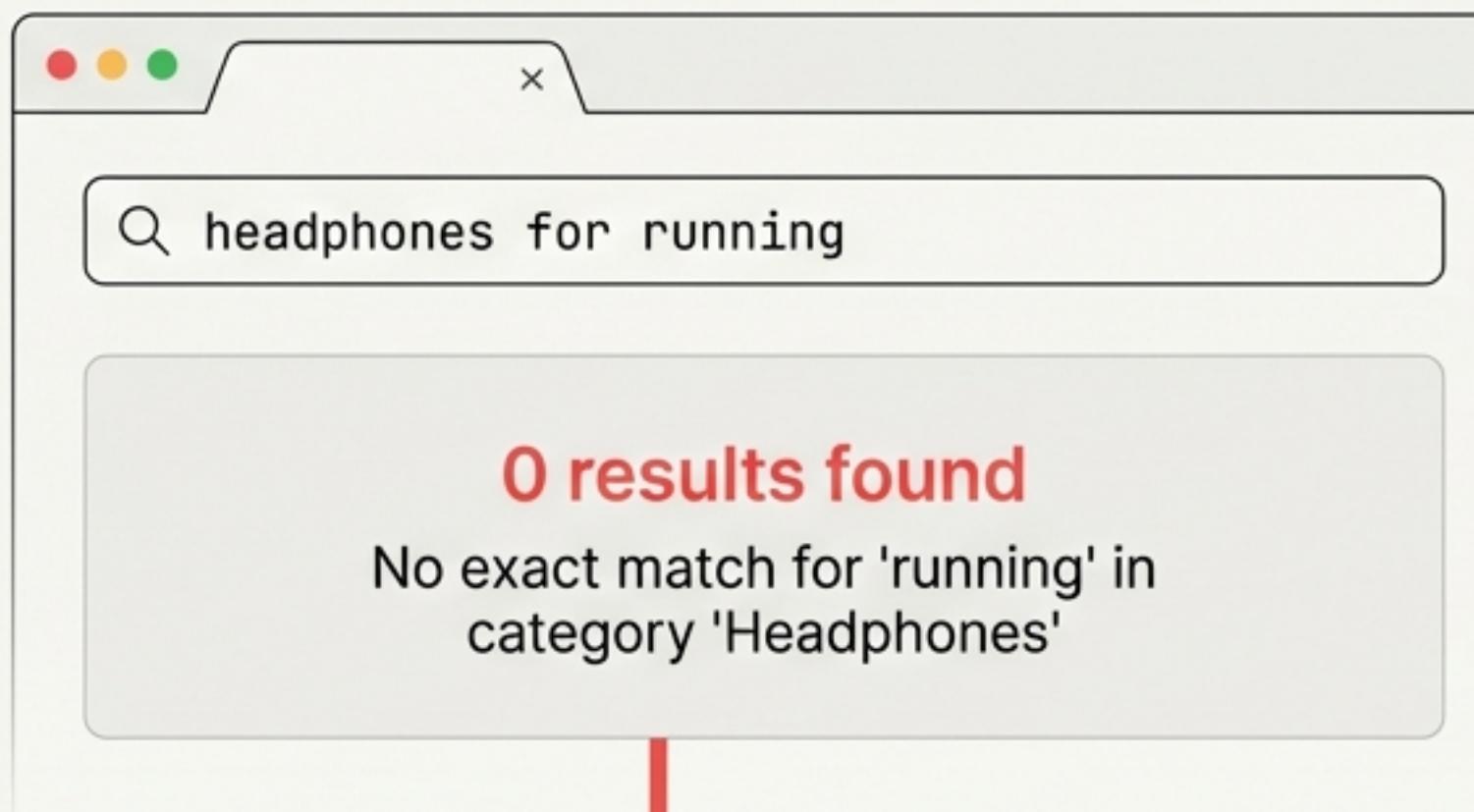
STACK: Python 3.10, FastAPI, React.js, PostgreSQL

EST. BUILD TIME: 2-3 Weeks (Fast Track)



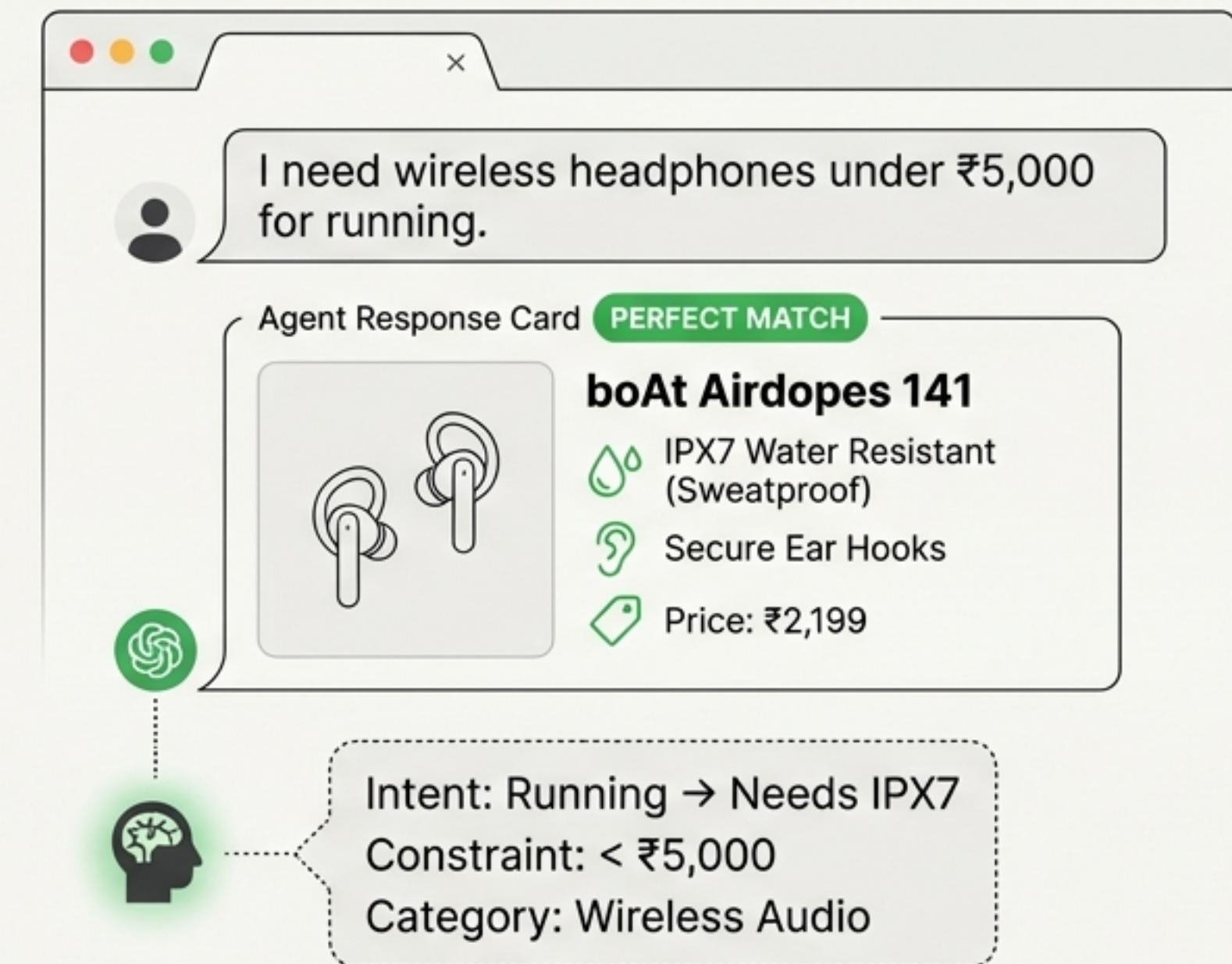
Moving From Keyword Search to Intent Understanding

Standard Keyword Search

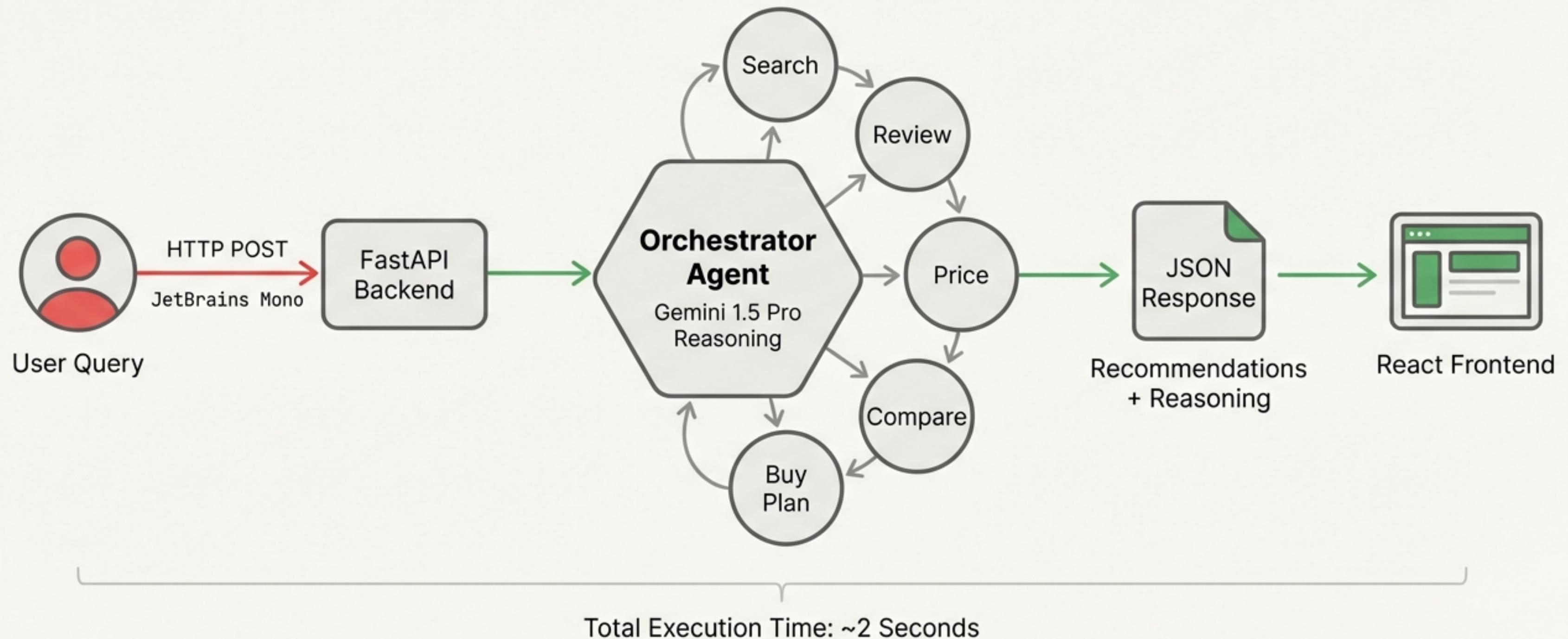


Fails to infer context
(Running = Sweatproof + Secure Fit)

Agentic Semantic Understanding



The Anatomy of an Intelligent Request



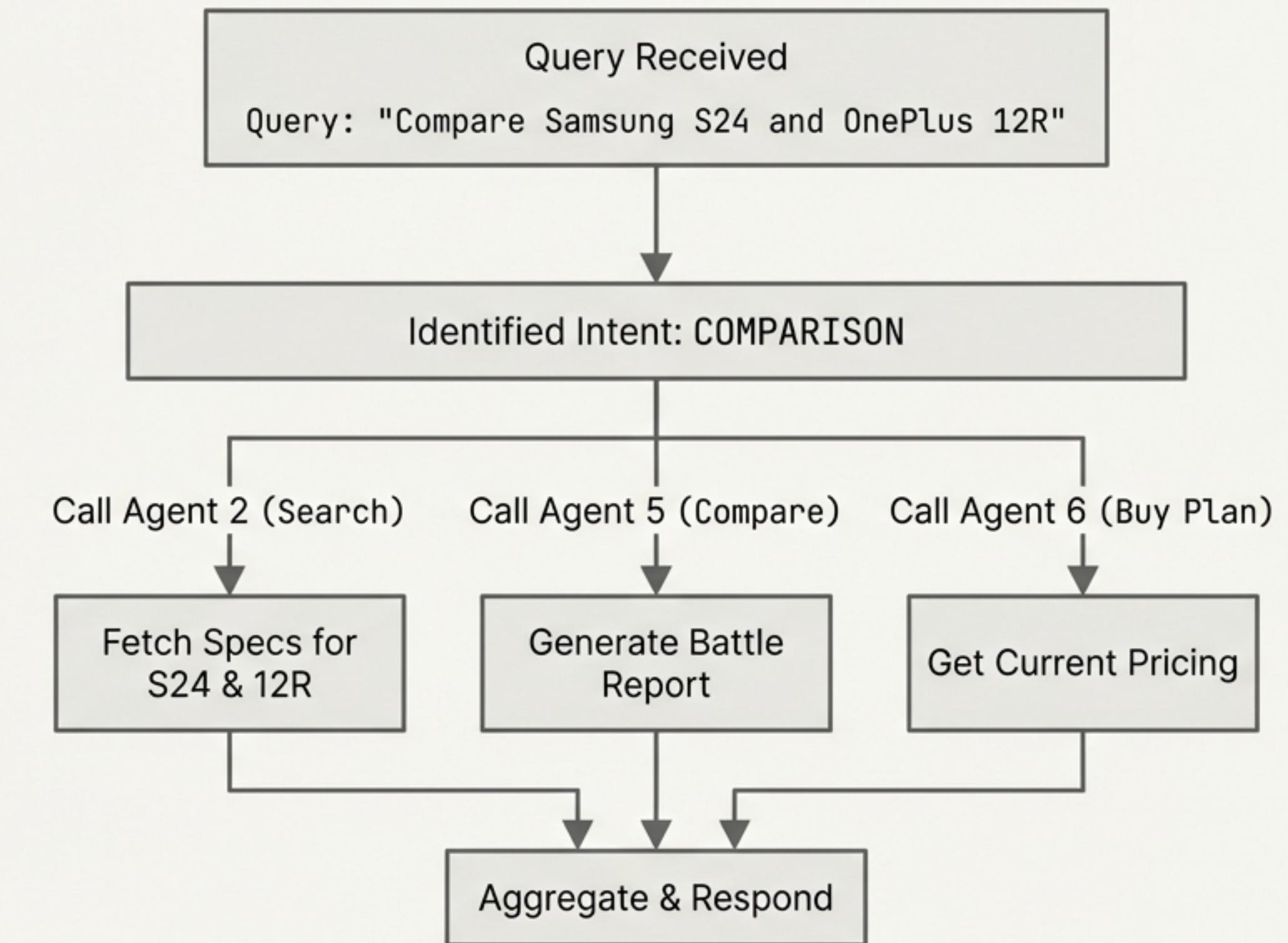
Six Specialized Agents, One Cohesive System



Agent 1: The Orchestrator & Reasoning Logic

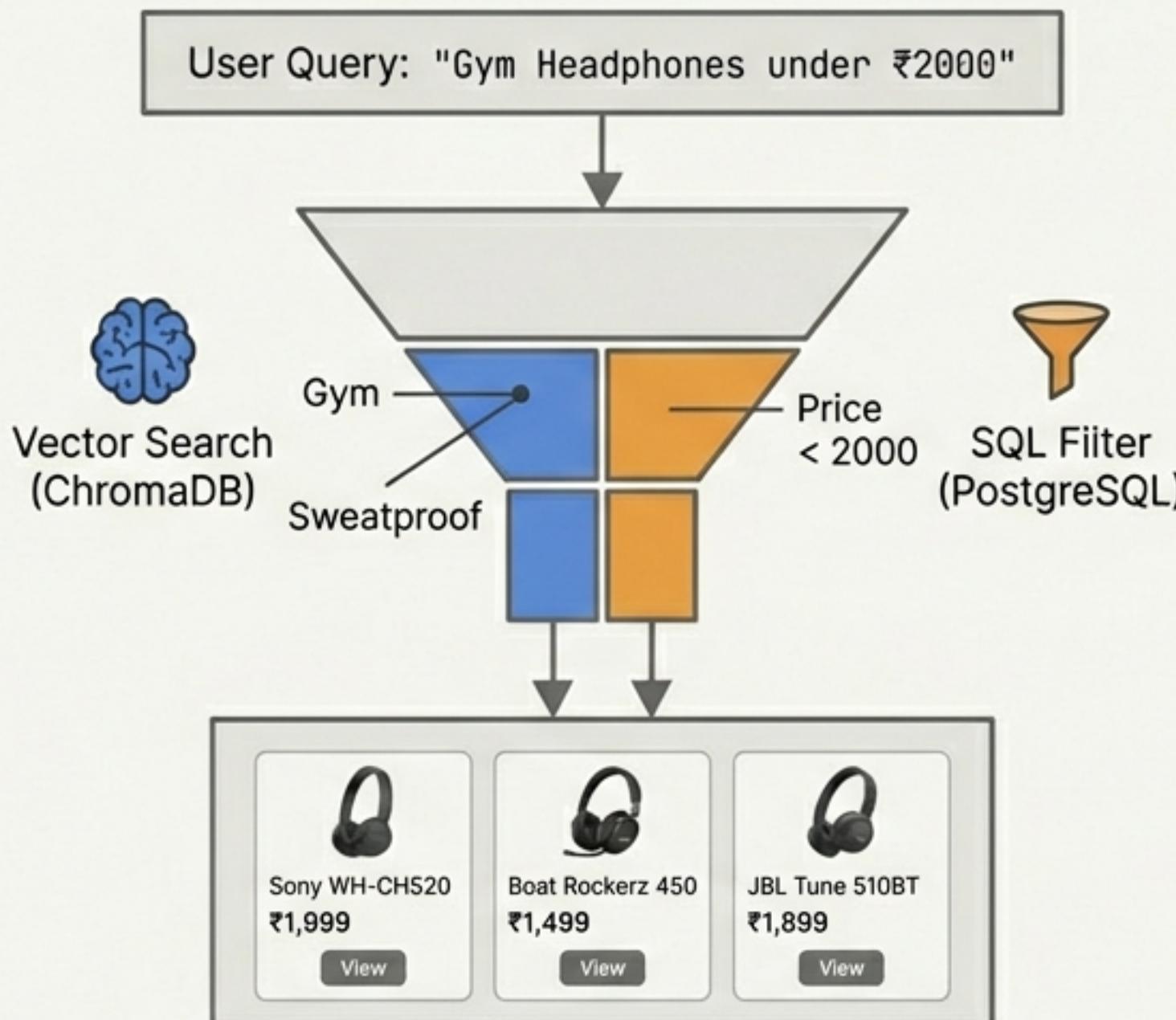
The Master Controller

- Role: Delegation & Planning
- Model: Gemini 1.5 Pro
- Input: Natural Language
- Output: Structured JSON Plan

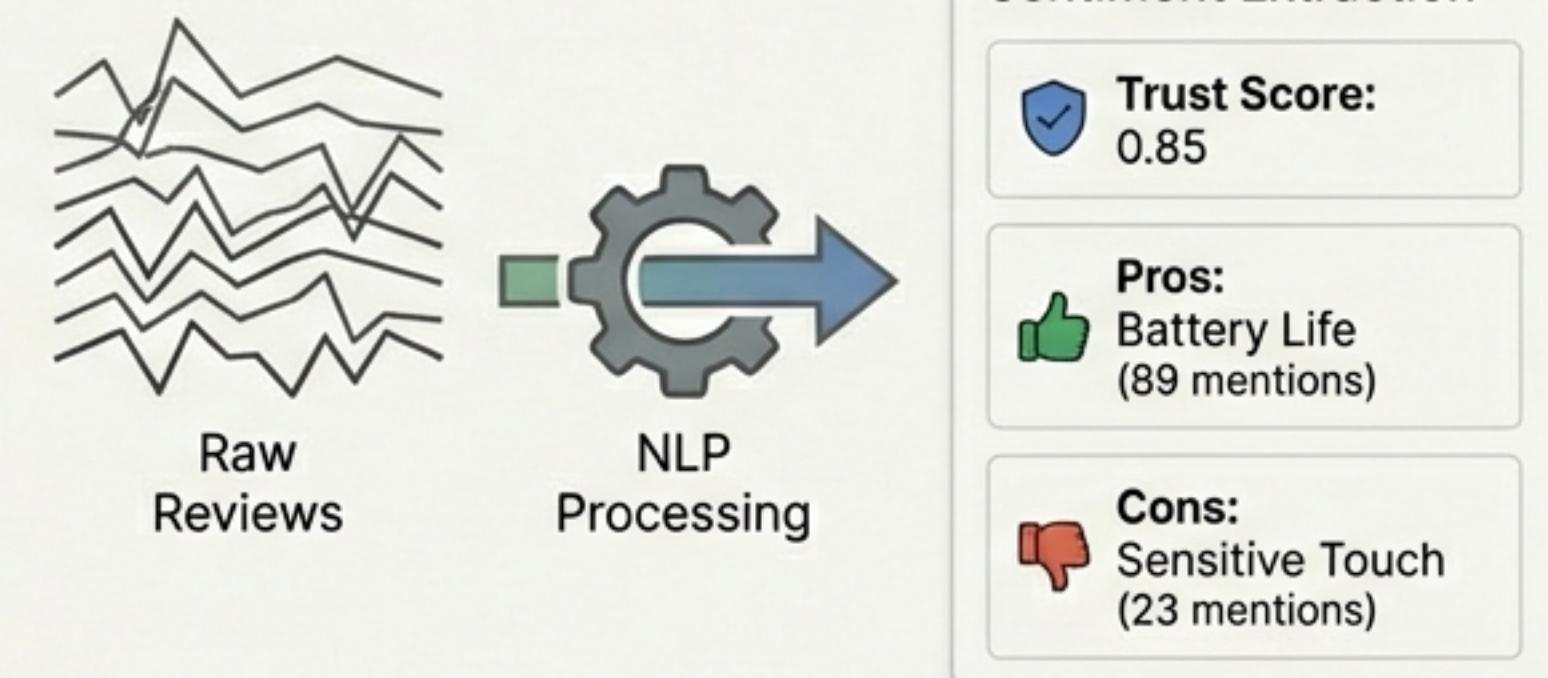


Discovery & Validation (Agents 2 & 3)

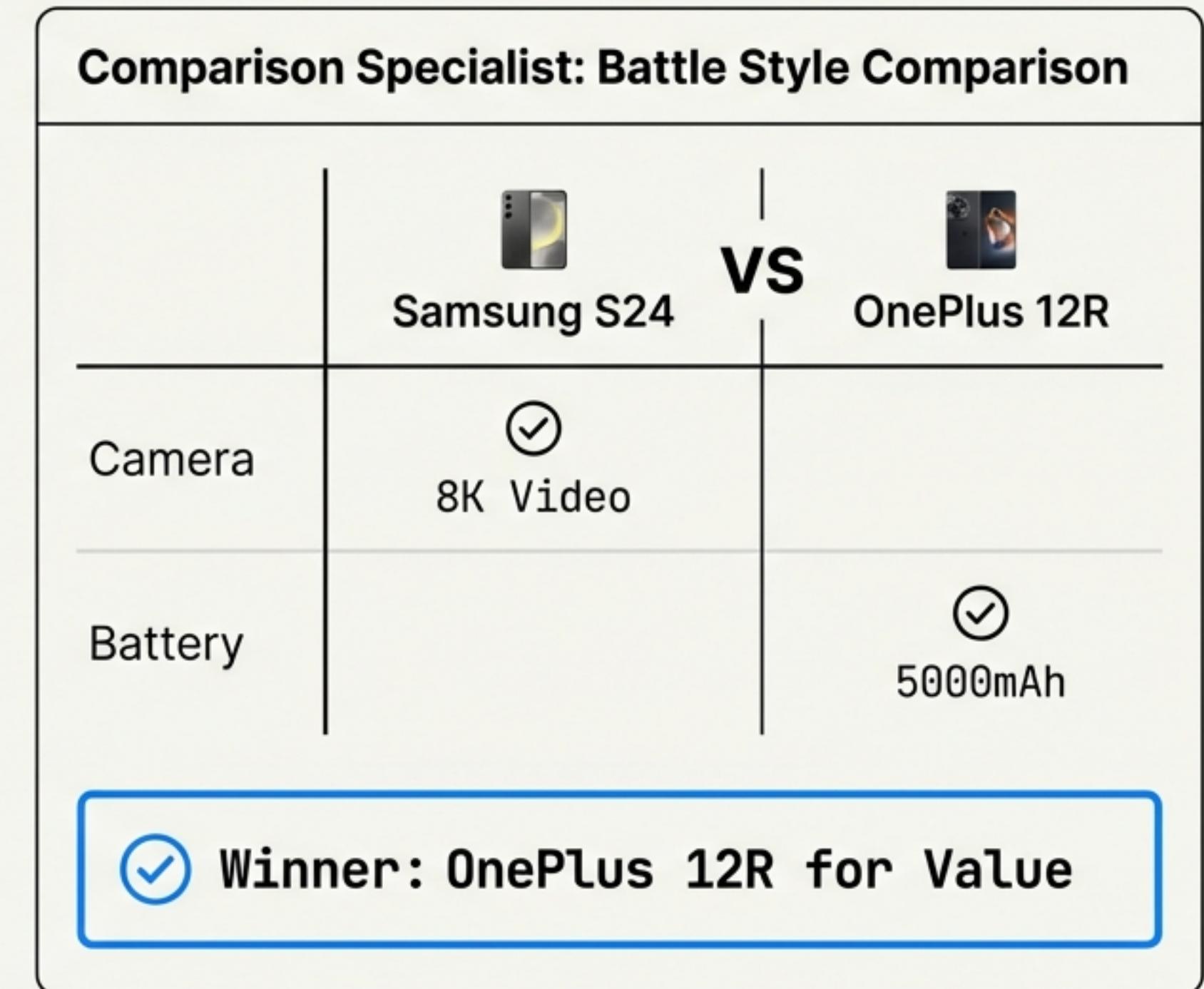
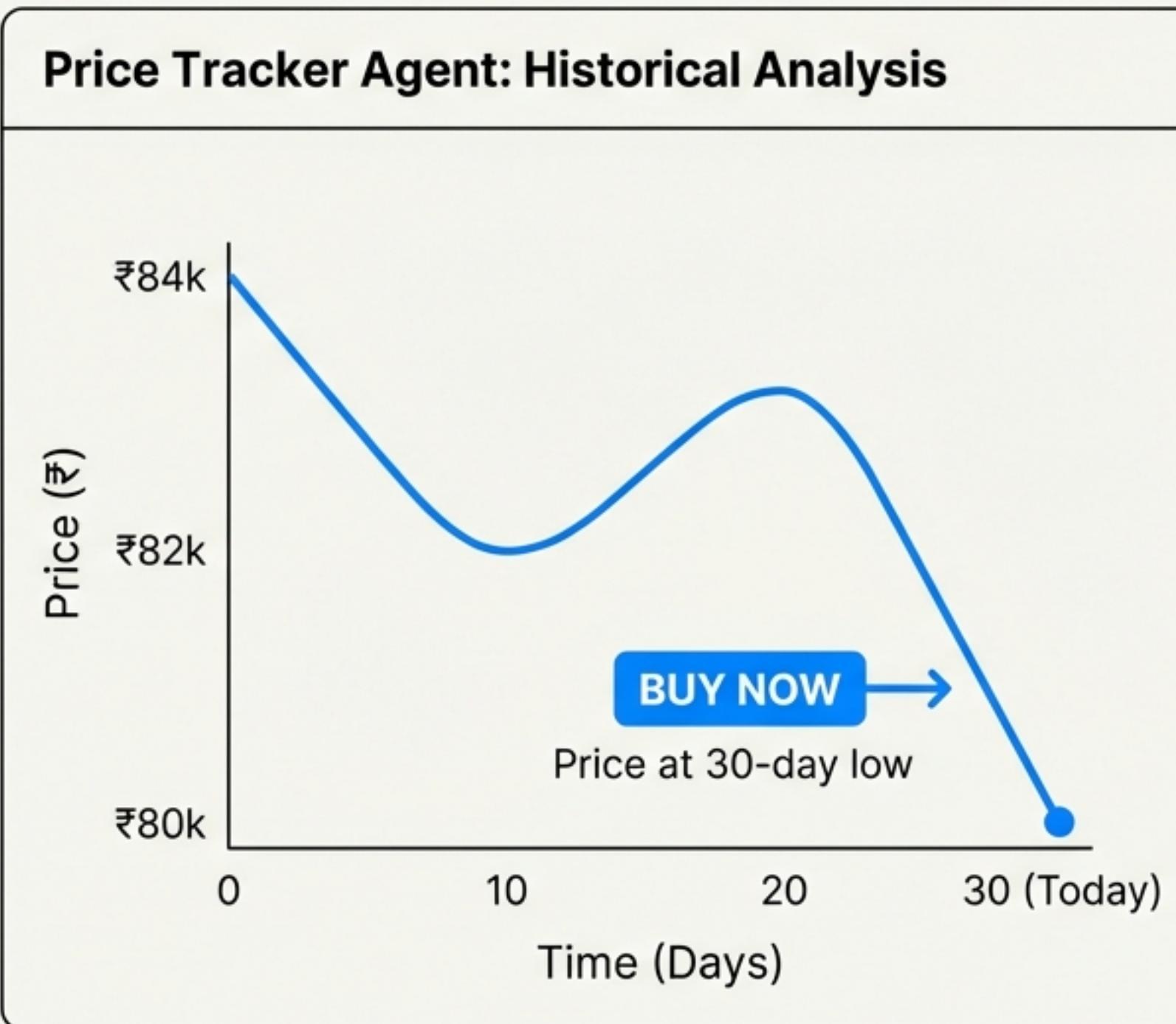
Product Search Agent: Hybrid Search Logic



Review Analyzer Agent: Sentiment Extraction



Valuation & Analysis (Agents 4 & 5)



Closing the Deal: The Buy Plan Optimizer (Agent 6)

Calculating the effective price, not just the sticker price.

Payment Optimization Logic

Pricing Breakdown

HP Victus 15

Base Price (MRP) ₹58,999

Instant Discount (HDFC Card) -₹3,000

Cashback (SBI Offer) -₹2,000

Effective Final Price ₹53,999

Recommendation Logic

Agent suggests:
"Use HDFC Card for
immediate discount" vs
"ICICI for No Cost EMI"
based on user preference.

Defining Agent Personality & Instructions

```
class PriceTrackerAgent(BaseAgent):
    def __init__(self):
        self.instructions = """
        You are a Price Tracker Agent specializing in deal detection.

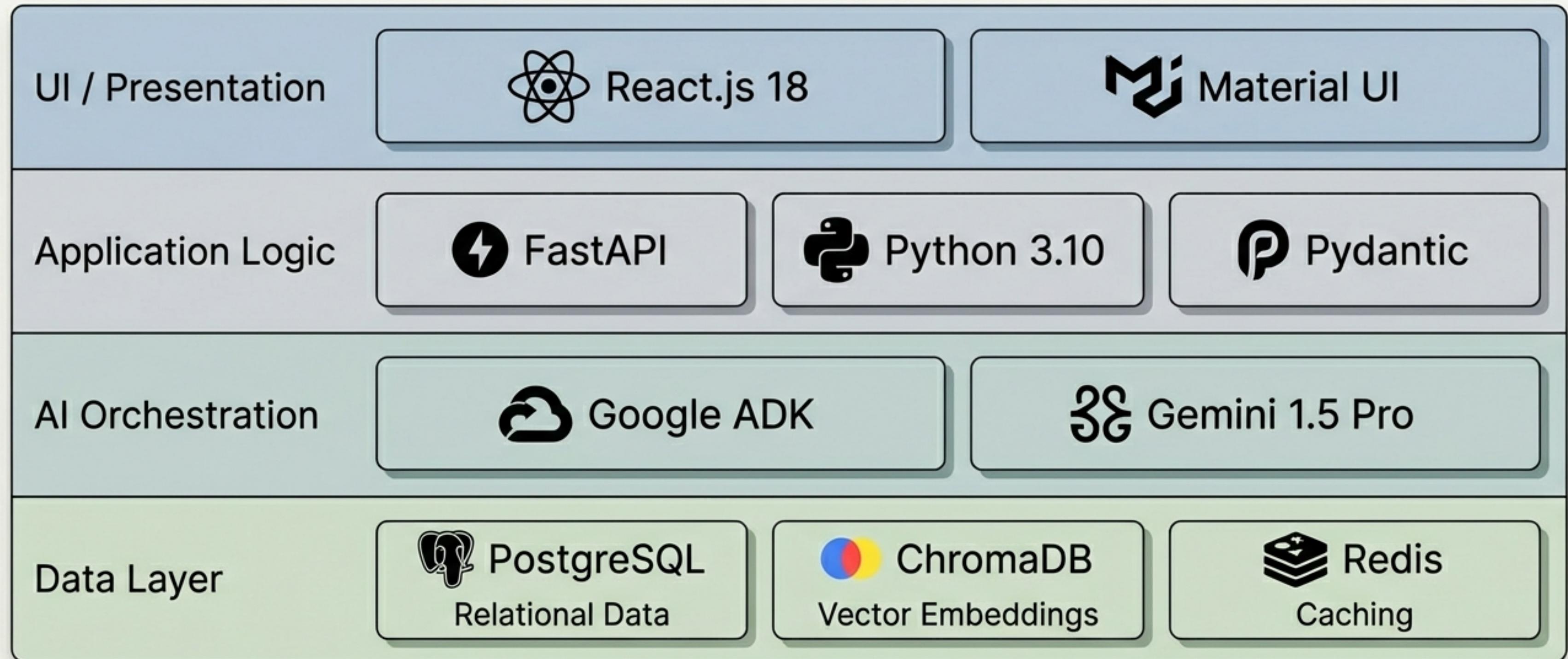
        Your capabilities:
        1. Analyze price history (30, 60, 90 days)
        2. Detect patterns (seasonal, event-based)
        3. Recommend BUY or WAIT decisions

        Signal Logic:
        - BUY: Price is near 30-day low OR >10% drop in 7 days.
        - WAIT: Price is trending up OR above 7-day average.
        """

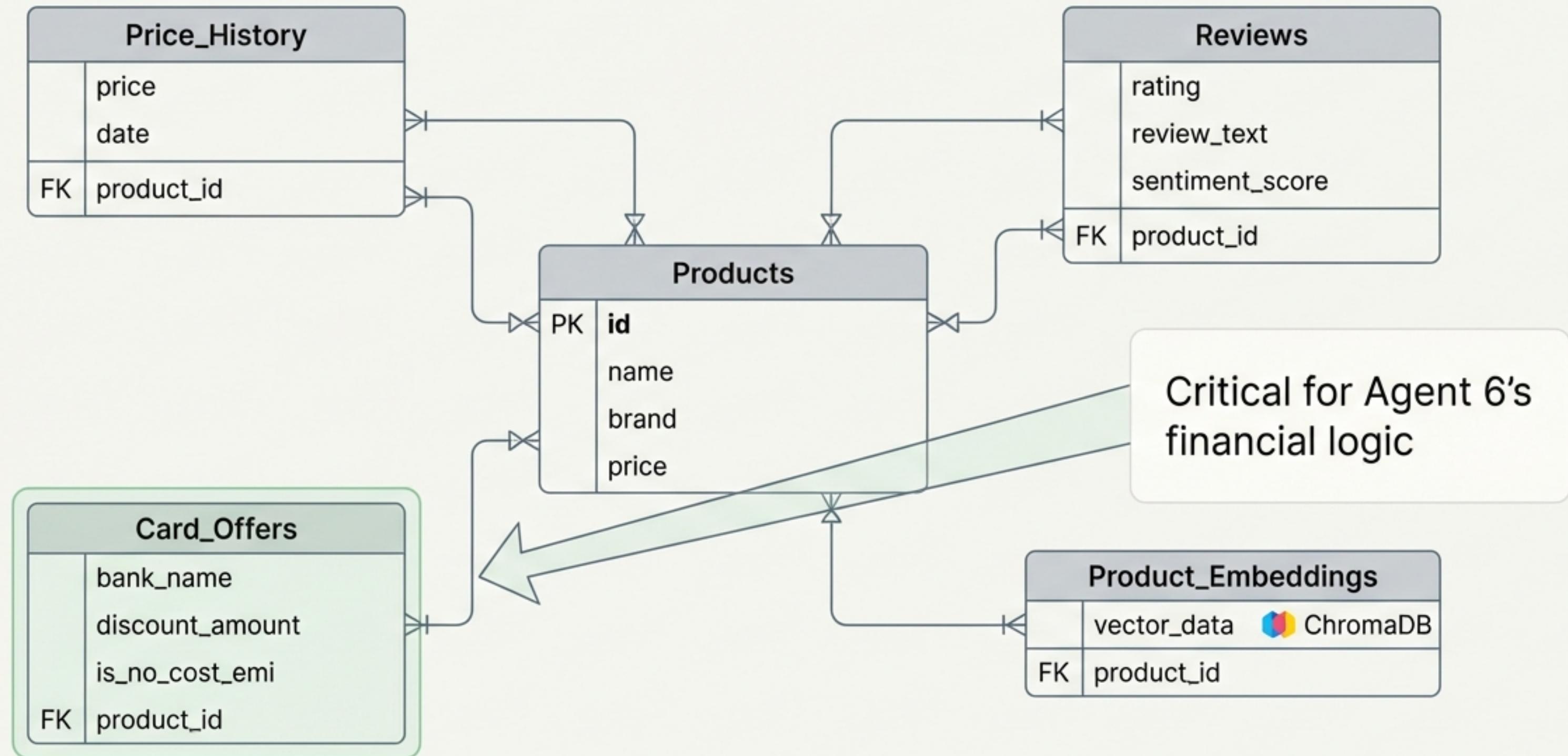
    self.tools = [get_price_history, check_active_deals]
```

Natural Language System Prompt

The Technical Stack



Data Architecture & Schema



Engineering for Production



Quality Assurance

- 85% Code Coverage via Pytest
- Resilient Error Handling Patterns



Monitoring

- Prometheus Metrics Collection
- Grafana Visualization Dashboards



Deployment

- Docker Containerization
- Kubernetes Orchestration



Security

- JWT Token Authentication
- Protected API Routes

Implementation Roadmap: The Foundation



Implementation Roadmap: Orchestration & Launch



The Future of E-Commerce is Agentic

