Architecture -

[User Prompt]

|

v

[LLM Agent (Gemini/GPT/LangChain)]

|

v

[Query Planner] ---> [Memory/Schema Embedding Store]

|

v

[MongoDB Query Generator]

|

v

[MongoDB Atlas]

|

v

[Results to Visualization Layer (e.g., Plotly/Chart.js/Streamlit)]

// Using Google

[User Prompt]

↓

[Gemini via Vertex AI API]

↓

[Tool Selection Logic: Aggregation, Atlas Search, Vector Search]

↓

[MongoDB Query Builder]

↓

[MongoDB Atlas]

↓

[Results → Visualized in Streamlit/React]

---------------------------------------------------------------------------

[Tool Selection] - SYSTEM PROMPT:

You are a data agent that selects the best MongoDB query type for a given natural language question. Your choices are:

- "aggregation" → for group, sort, average, trend, or summary-type queries.

- "atlas\_search" → for keyword or text matching.

- "vector\_search" → for semantic similarity or "like this" queries.

Only return the tool name. No explanation.

USER: Show me the average purchase amount grouped by month.

→ aggregation

USER: Find reviews that mention “sustainable materials”.

→ atlas\_search

USER: Show me products similar to Apple AirPods.

→ vector\_search

| **Approach** | **Use Case** | **Pros** | **Cons** |
| --- | --- | --- | --- |
| Rule-Based | MVP, simple use | Fast, easy | Rigid, fails on nuanced queries |
| Gemini-Powered | Realistic prompts | Flexible, easy to expand | Requires LLM tokens |
| Agent + Function Calling | Complex queries | Fully autonomous | Slightly more infra needed |
|  |  |  |  |

| **Prompt** | **What it should do** |
| --- | --- |
| “Find articles like the one about Elon Musk’s Mars plans.” | Vector search |
| “Search for articles that mention climate change and policy.” | Atlas Search |
| “Top 5 topics written about by New York Times in 2020” | Aggregation |