MEX Assistant

Presented by Wi-Five

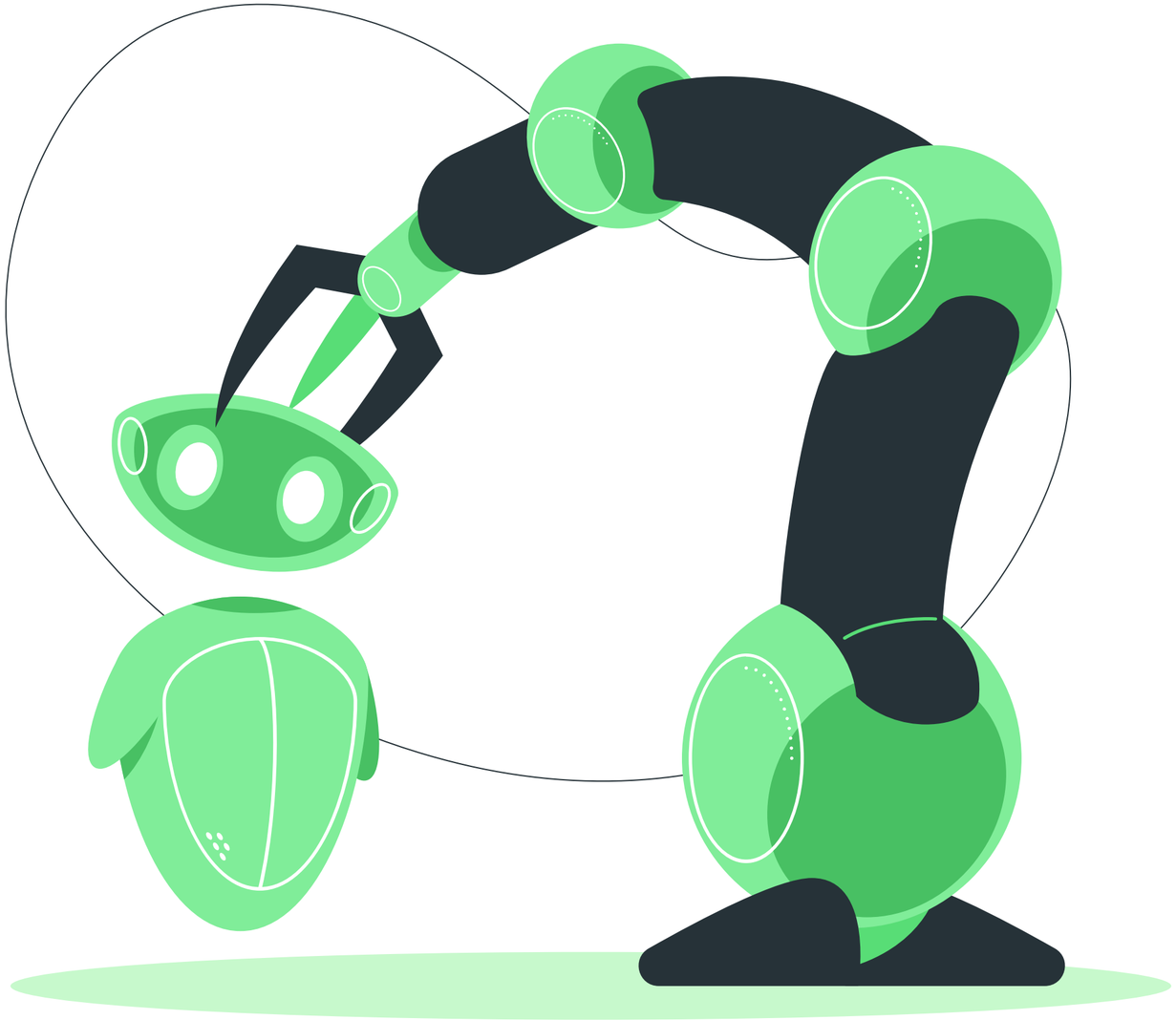
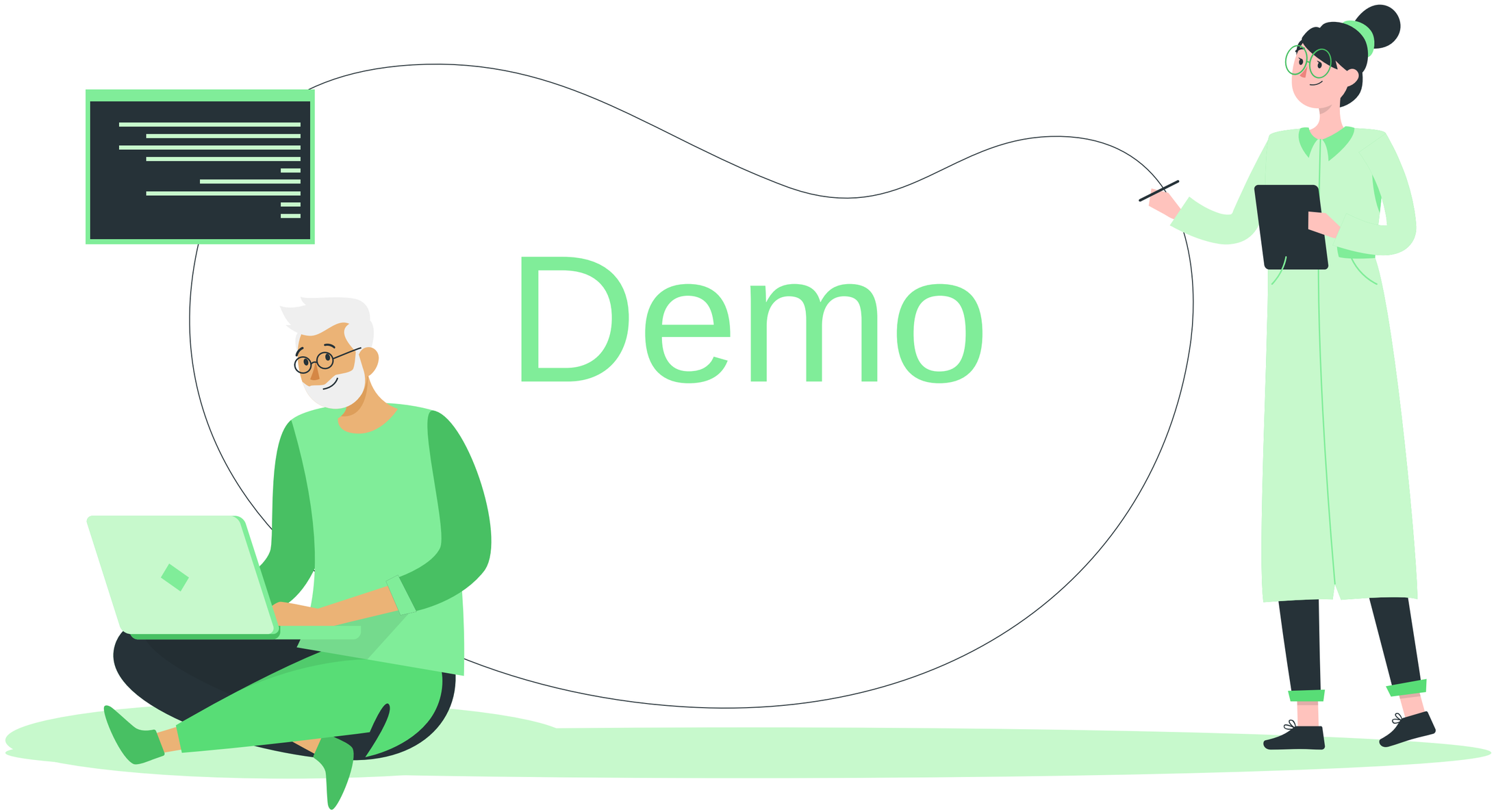
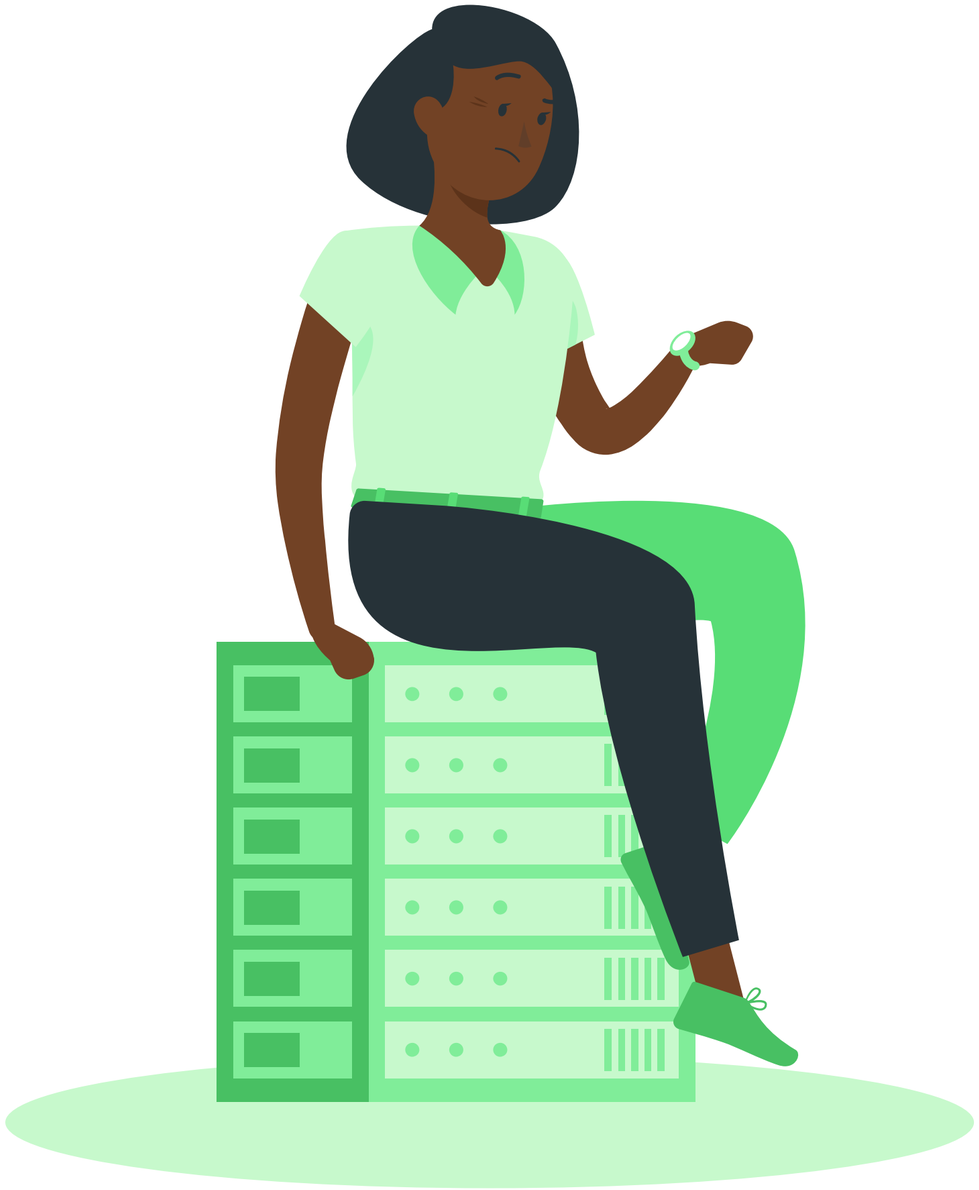


Table of Contents

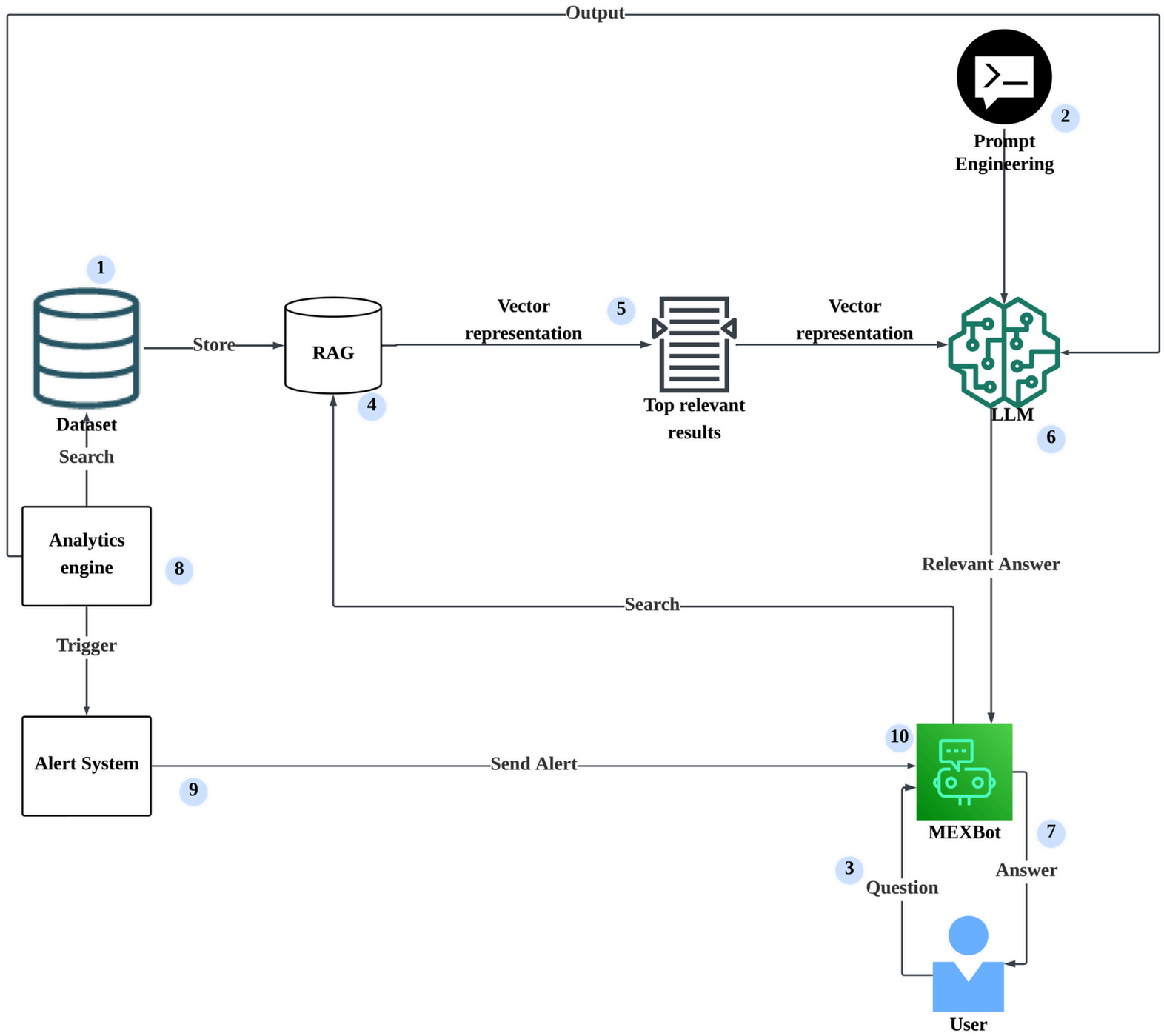
Demo





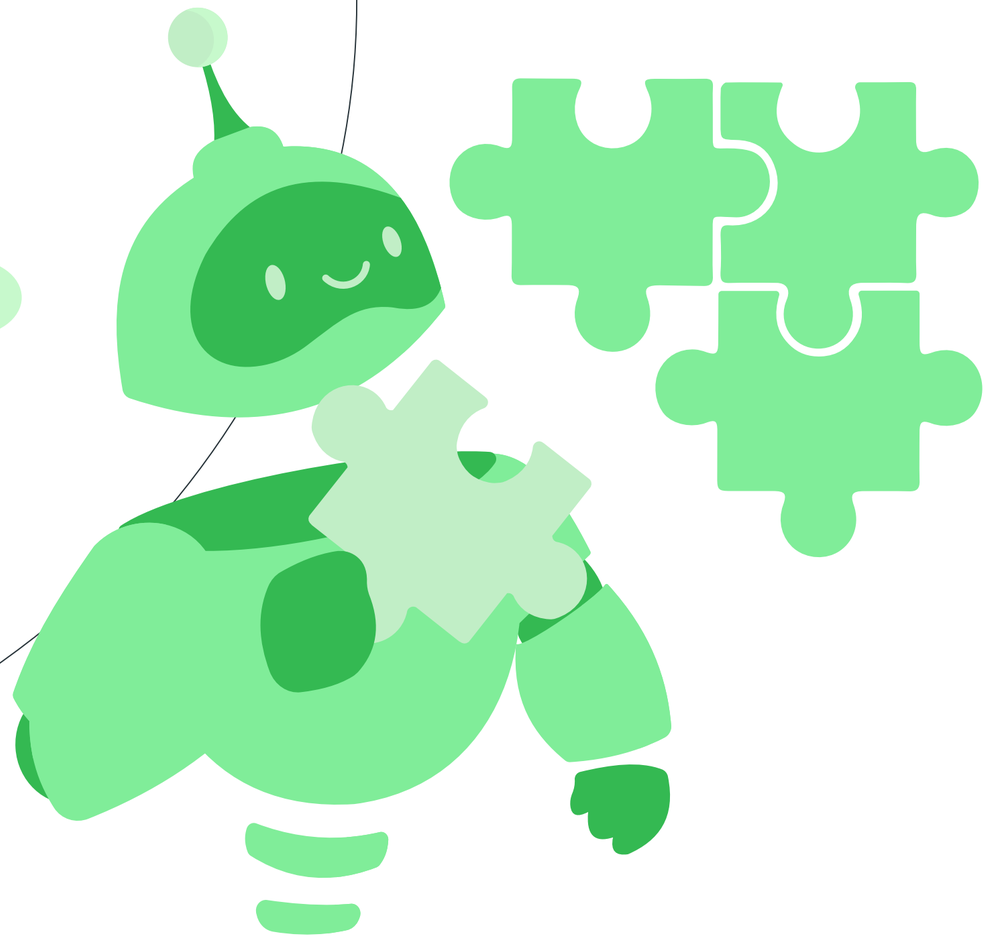
Solution Architecture

Architecture Diagram



1. Stores data.
2. Prompt optimizes user queries.
3. User asks question to MEXBot.
4. Searches relevant info from the data.
5. Returns top relevant document.
6. LLM generates the answer.
7. MEXBot replies answer to user.
8. Monitors and analyzes data.
9. Sends alerts if triggered.
10. Alerts sent to MEXBot.

Technical Infrastructure



Technical Infrastructure

MEXBot respond intelligently to user queries using datasets through two methods:

- **Retriever-Augmented Generation (RAG)**
- **Large Language Model (LLM)**

Backend services and APIs used

- **10 steps**

Backend services and APIs used

1 Data Ingestion

The system ingests datasets including both **structured** (e.g., sales.csv, inventory.csv) and **unstructured** (e.g., merchant_guides.txt) data.

- **Structured** data is used by the ***Analytics Engine***.
- **Unstructured** text is converted into vectors using an embedding model and stored in the ***RAG vector*** store for semantic retrieval.

Backend services and APIs used

2

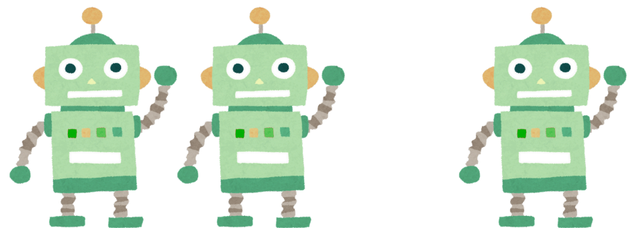
Structured Prompt

A structured prompt that guides the LLM to generate focused and relevant responses.

3

User Query

The user asks a question through MEXBot, which forwards it to the backend for processing.

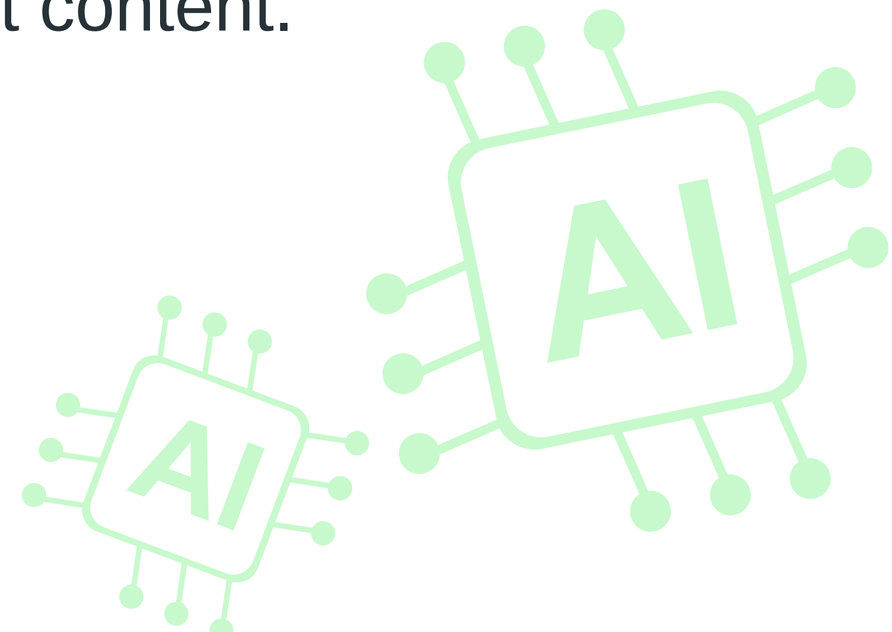


Backend services and APIs used

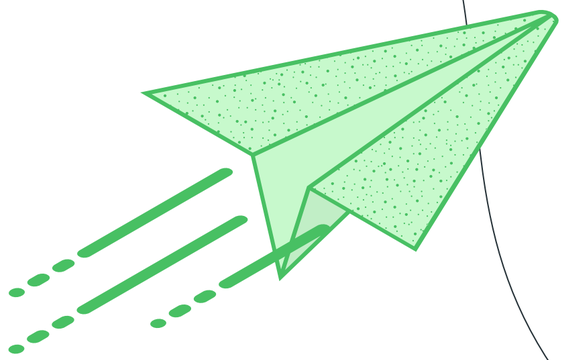
4 Semantic Query Handling

For semantic questions:

- MEXBot sends the question to RAG.
- RAG searches the vector store for relevant content.



Backend services and APIs used



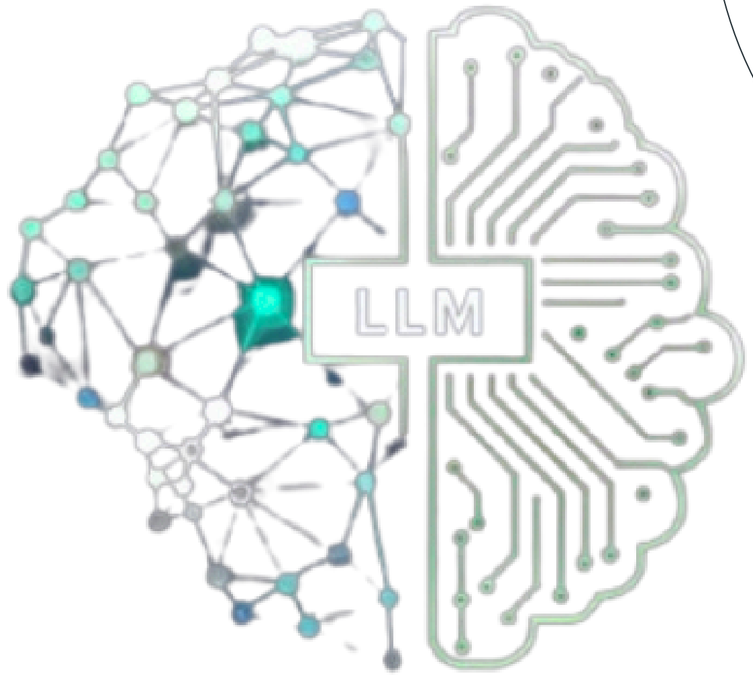
5 RAG Retrieval

- MEXBot sends the semantic query to RAG.
- RAG performs a similarity search on the vector store.
- Retrieves top matching document chunks relevant to the user’s query.
- These chunks are used to construct the final prompt for the LLM.

Backend services and APIs used

6 LLM Processing

- The prompt is sent to the LLM which generates a response.



7 Response Delivery

- MEXBot receives the LLM’s answer and presents it clearly to the user.

Backend services and APIs used

8 Analytics Engine for Structured Queries

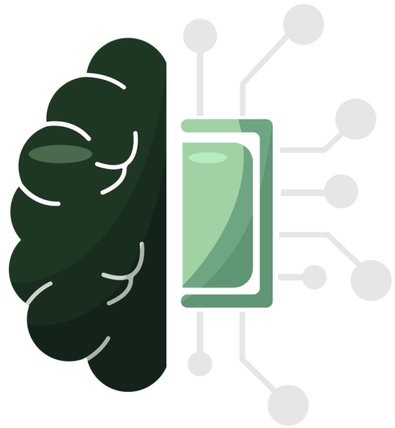
- For structured questions:
  - The ***Analytics Engine*** processes CSV datasets *(e.g., calculates KPIs: sales trends, inventory levels)*.
  - Runs periodically to monitor metrics.
  - If anomalies/threshold breaches are detected *(e.g., low stock)*, results are passed to the ***Alert System*** → pushes notifications to MEXBot.

Backend services and APIs used

9 Alert System Activation

- The ***Alert System*** is triggered by flagged outputs from the ***Analytics Engine***.
- When a KPI or threshold breach is detected:
    1. Analytics Engine passes results to the Alert System.
    2. Alert System pushes alerts to MEXBot.

Backend services and APIs used



10 MEXBot's Response and Intent Detection

MEXBot presents responses/alerts to users via chat interface:

- Uses **natural language** and visual cues.
- Uses **Intent Detection** classifies user queries into two types:
  - **Structured Intent** *("What are my sales this week?")*
    - Sent to the Analytics Engine for data-driven answers.
  - **Open-ended/Semantic Intent** *("How can I improve delivery service?")*
    - Sent to the RAG + LLM pipeline for more complex responses.

Thank you