Decision Log

Project: Monday.com Business Intelligence Agent
Candidate: Sunkeerth
Date: February 27, 2026

---

1. Introduction

The goal of this project was to build an AI agent that answers founder-level business questions by fetching live data from Monday.com boards (Deals and Work Orders). The agent must handle messy data, provide insights on revenue, pipeline health, and sector performance, and show a trace of every action.

To enhance usability, I also added voice output and simple data visualizations, making the agent interactive and easier to use.

2. Technology Stack

Layer	Technology Choice	Reason
Backend	Python + FastAPI	FastAPI is fast, async-friendly, and ideal for building REST APIs. Python also provides strong data processing libraries like pandas.
Data Handling	Pandas	Used for cleaning data, parsing dates, handling missing values, and processing currency fields efficiently.
Frontend	HTML, CSS, JavaScript	Lightweight and fast without framework overhead. Used Chart.js for charts and Web Speech API for voice output.
Hosting	Render (Backend), Netlify (Frontend)	Both platforms provide free hosting with GitHub integration and automatic deployment.

3. Data Cleaning Approach

The provided sample data was intentionally inconsistent and messy.

Date Handling:
Dates appeared in multiple formats such as:

- `2025-12-31`
- `Feb 25`
- `31-12-2025`

A custom `parse_date()` function was implemented to try multiple date formats and convert them into a standard datetime object. If parsing fails, the function returns `None`.

Currency Handling:
Currency values sometimes included:

- Commas
- “Rs” symbols
- Empty fields

The clean_currency() function removes non-numeric characters and converts values into floats. Missing values are converted to 0.0.

Missing Values:

- Missing dates → None
- Missing amounts → 0.0

Dynamic Column Detection:
Column names differed across boards (e.g., "Sector" vs "Sector/service").
To solve this, I implemented keyword-based column detection. The system searches column names for keywords such as:

- sector
- status
- value
- date

This ensures the agent remains functional even when naming conventions differ.

4. Query Understanding

A rule-based keyword extraction approach was implemented.

Example:
If a query contains keywords like **"pipeline"** and **"sector"**, the agent extracts the sector name (e.g., "mining") and filters open deals within that sector.

This method is sufficient for predictable business queries and ensures fast execution.

Extensibility:
The system is designed so that keyword logic can later be replaced with an LLM-based query parser for more complex natural language understanding. The trace panel can also log future LLM calls.

---

5. Live API Integration

Each query triggers **live API calls** to Monday.com with no caching.

Workflow:

1. Backend sends GraphQL request to Monday.com API.
2. Fetches all items from Deals and Work Orders boards.
3. Cleans and processes data.
4. Generates response.

The frontend trace panel displays each step, for example:

- "Fetching Deals board…"
- "Fetched 346 deals."
- "Interpreting query…"

This ensures full transparency and confirms real-time data usage.

6. Error Handling

Robust error handling ensures reliability:

- Invalid API key or fetch failure → friendly error message displayed.
- Missing columns → handled safely with default values.
- Unsupported queries → agent suggests valid queries.
- All issues logged in trace panel for debugging visibility.

7. Innovation: Voice Output and Charts

Voice Output

The Web Speech API is used to read answers aloud.

Benefits:

- Improves accessibility
- Enables hands-free usage
- Provides modern conversational experience

Implementation:
After receiving the backend response, the frontend calls:
`window.speechSynthesis.speak()`

Voice activates after user interaction (browser compatibility requirement).

Charts Visualization

Chart.js is used to display visual insights.

Example:
When querying pipeline by sector, a bar chart shows deal stage distribution.

Benefits:

- Faster understanding of insights
- Visual representation of business data
- Demonstrates aggregation capability

Charts appear only when meaningful data is available.

8. Challenges Faced :

Challenge	Solution
Column name mismatches	Implemented keyword-based column detection
Inconsistent date formats	Built flexible multi-format date parser
Currency symbols and commas	Cleaned values before numeric conversion
API rate limits	Added fallback “try again later” message
Browser voice autoplay restrictions	Voice triggers after user interaction

9. Assumptions Made

- Sector names remain mostly consistent across boards.
- “Pipeline” refers to deals marked Open or On Hold.
- Revenue is calculated from Won deals.
- Queries will be asked in English using common business terms.
- Boards contain sufficient sample data.

10. Conclusion

The AI agent successfully meets all assignment requirements:

- Live Monday.com API integration with trace visibility
- Strong data cleaning and resilience
- Founder-level business query understanding
- Revenue, pipeline, and sector analytics
- Transparent action trace panel
- Interactive UI with voice and charts

All components are deployed and publicly accessible.
The system is production-ready, modular, and easily extendable for future AI enhancements.

This project demonstrates the ability to design and deploy a complete AI-driven business intelligence agent within a short timeframe.