Google Developer Groups
Cloud · New Delhi

Turgon

Team: Zero Kelvin

Round I

IDEA SUBMISSION

HACKFEST 2.0

Google Developer Groups
Cloud·New Delhi

Turgon

Team Details:

S no.	Team Member Name	Team Member Role	Team Member Mail ID
1.	*Jakaria Ahmed*	*Data Engineer*	ahmed.ka.workmail@gmail.com
2.	Arunoday Singh	*Software Developer*	arunodaysingh137@gmail.com
3.	Zakir Ahmed	*Software Developer*	ahmed.jakaria080800@gmail.com
4.			

HACKFEST 2.0

Track Chosen:

Track Chosen: Intelligent Data Dictionary Agent

Problem Description:

1. **The Problem:** Most companies struggle with messy **legacy** databases where documentation is either missing or years out of date. Technical labels often make no sense to the people(biz folks) who need them.
2. **Why It Matters:** This creates a "data graveyard." Teams waste 80% of their time just deciphering data instead of using it, causing slow decisions and expensive errors.

Google Developer Groups
Cloud·New Delhi

Turgon

Proposed Solution / Idea Overview:

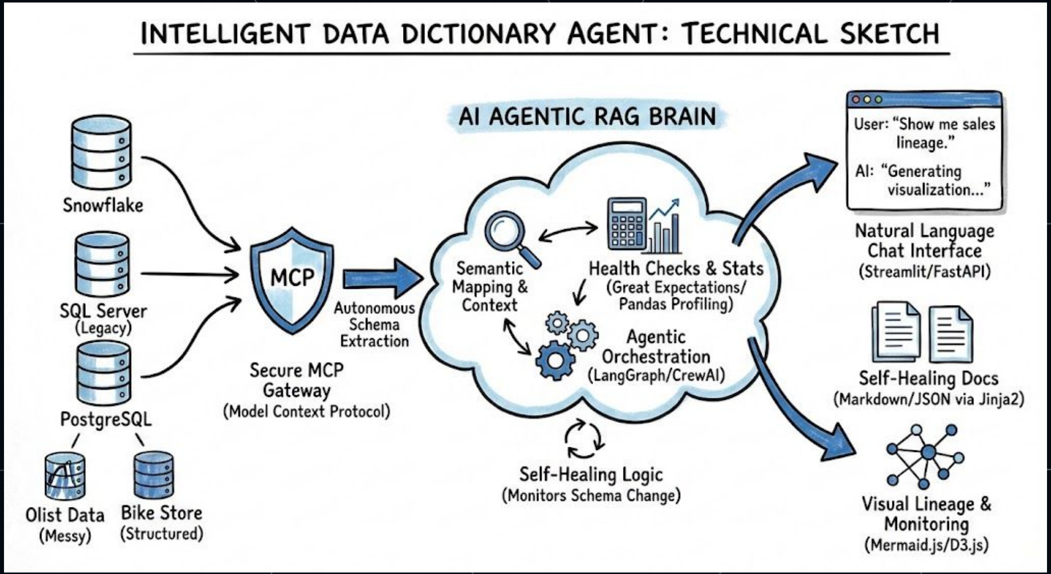
Autonomous Extraction & MCP: We propose an agent that connects to **legacy** and modern databases to automatically "read" their entire structure. By using **MCP** for DBs ,wherever possible, we expose database schemas directly to our AI, giving agents the deep context needed to map complex relationships and constraints without any manual help.

AI Intelligence & Health Checks: The system performs real-time "health checks" using statistical analysis to spot issues like missing data or outdated records. It uses AI to translate cryptic code into clear business summaries, turning confusing technical metadata into trustworthy, easy-to-understand institutional knowledge.

Agentic RAG & Chat Interface: The final layer is an **Agentic RAG** framework providing a conversational chat interface. Instead of writing SQL, users ask questions in plain English to find data or get query suggestions. This ensures the "data graveyard" becomes a searchable, self-healing knowledge base for everyone.

HACKFEST 2.0

Proposed Solution / Idea Overview:


INTELLIGENT DATA DICTIONARY AGENT: TECHNICAL SKETCH
Snowflake
SQL Server (Legacy)
PostgreSQL
Olist Data (Messy)
Bike Store (Structured)
MCP
Secure MCP Gateway (Model Context Protocol)
Autonomous Schema Extraction
AI AGENTIC RAG BRAIN
Semantic Mapping & Context
Health Checks & Stats (Great Expectations/ Pandas Profiling)
Agentic Orchestration (LangGraph/CrewAI)
Self-Healing Logic (Monitors Schema Change)
User: "Show me sales lineage."
AI: "Generating visualization..."
Natural Language Chat Interface (Streamlit/FastAPI)
Self-Healing Docs (Markdown/JSON via Jinja2)
Visual Lineage & Monitoring (Mermaid.js/D3.js)

(Generated with Gemini)

Google Developer Groups
Cloud · New Delhi

Turgon

Additional Notes:

Real-World Validation: We are stress-testing the agent using the **Olist E-Commerce dataset** to handle messy **legacy** data (nulls, inconsistent categories) and the **Bike Store SQL database** to validate strict relational mapping and foreign key extraction.

The Tech Stack:

- **Orchestration:** Python-based **LangGraph** for the Agentic RAG framework.
- **Context Layer: MCP (FastMCP),** to provide a secure, standardized bridge between the LLM and database schemas.
- **Quality Engine: Pandas Profiling** for calculating statistical metrics like Z-Scores (outliers) and Shannon Entropy (data diversity).
- **Frontend:** A **Streamlit** web interface for the natural language chat.

Self-Healing & Exports: The system monitors `INFORMATION_SCHEMA` to detect schema drifts automatically. It uses **Jinja2 templating** to generate documentation in Markdown and JSON, ensuring the "map" is always fresh and version-controlled.

Proactive Data Lineage: Beyond simple chat, the agent will visualizes table relationships(object relationship) using **Mermaid.js** or **D3.js**, allowing users to see exactly how data flows through their organization. Yet to be explored.

HACKFEST 2.0