

# Sentinel: Fraud Detection AI Workbench

---

*Comprehensive System Documentation*

**Architecture, Integrated Design, and User Manual**

# Introduction to Fraud detection AI Workbench (version pro)

## Contents

1. Introduction of Applications .....	5
2. Integrated Design.....	5
2.1 System Architecture & Data Flow.....	5
2.2 UI Design Strategy.....	6
3. Security Design .....	6
3.1 Login Implementations & RBAC.....	6
3.2 Data Security (SQL Anti-Injection).....	6
3.3 Demo.....	7
3.3.1.1 Login Way 1 .....	7
3.3.1.2 Login Way 2 .....	7
3.3.2 Register .....	7
3.3.3 Admin Console – User Management .....	8
3.3.4 Admin Console - Role Management.....	8
4. UI1 - Home .....	8
4.1 UI Design.....	8
4.2 User Manual.....	8
4.3 Logic Workflow .....	9
4.4 Demo.....	9
5. UI2 - Data Hub .....	9
5.1 UI Design.....	9
5.2 User Manual.....	9
5.3 Logic Workflow .....	10
5.4 Demo.....	10
6. UI3 - SQL RAG .....	15
6.1 UI Design.....	15
6.2 User Manual.....	16
6.3 Logic Workflow .....	16
6.4 Demo.....	16
7. UI4 - Graph RAG.....	18
7.1 UI Design.....	18
7.2 User Manual.....	19
7.3 Logic Workflow .....	19

## Introduction to Fraud detection AI Workbench (version pro)

7.4 Demo.....	19
8. UI5 - Multimodal RAG .....	22
8.1 UI Design.....	22
8.2 User Manual.....	22
8.3 Logic Workflow .....	22
8.4 Demo.....	23
9. UI6 - Trends and Insights .....	28
9.1 UI Design .....	28
9.2 User Manual.....	28
9.3 Logic Workflow .....	28
9.4 Demo.....	28
10. UI7 - ML Workflow .....	30
10.1 Functionality Design.....	30
10.2 UI Design .....	30
10.3 User Manual .....	30
10.4 Logic Workflow.....	31
10.5 Demo .....	31
11. UI8 - LLM Fine Tuning.....	35
11.1 Functionality Design.....	35
11.2 UI Design .....	35
11.2 User Manual .....	35
11.3 Logic Workflow.....	36
11.4 Demo .....	36
12. UI9 - API Interaction Hub.....	41
12.1 UI Design .....	41
12.2 User Manual .....	41
12.3 Logic Workflow.....	41
12.4 Demo .....	41
13. UI10 - Admin Console.....	44
13.1 UI Design .....	44
13.2 User Manual .....	44
13.3 Logic Workflow.....	44
13.4 Demo .....	44

## Introduction to Fraud detection AI Workbench (version pro)

14. Show Demo .....	48
14.1 Description .....	48
14.2 Steps .....	48
14.3 Impacts to the System.....	48
15. Load Demo Data.....	49
15.1 Description .....	49
15.2 Steps and Location .....	49
15.3 Impacts to the System.....	49
15.4 Demo .....	50
16. Dependency References .....	51
16.1 FastAPI .....	51
16.2 Streamlit .....	51
16.3 SQLAlchemy .....	51
16.4 Neo4j (urllib3/neo4j-driver) .....	51
16.5 FAISS .....	51
16.6 SQLGlot.....	52
16.7 PyVis / NetworkX .....	52
16.8 BCrypt.....	52
16.9 Ollama.....	52
17. Glossary .....	52

# Introduction to Fraud detection AI Workbench (version pro)

## 1. Introduction of Applications

Sentinel Fraud AI Workbench Pro is a comprehensive, enterprise-grade platform engineered to detect, investigate, and mitigate financial fraud. It unifies Generative AI and traditional Machine Learning within a single, cohesive interface. The platform serves as a central hub for investigators, data scientists, and administrators to uncover hidden fraud rings, analyze massive datasets, and fine-tune models safely.

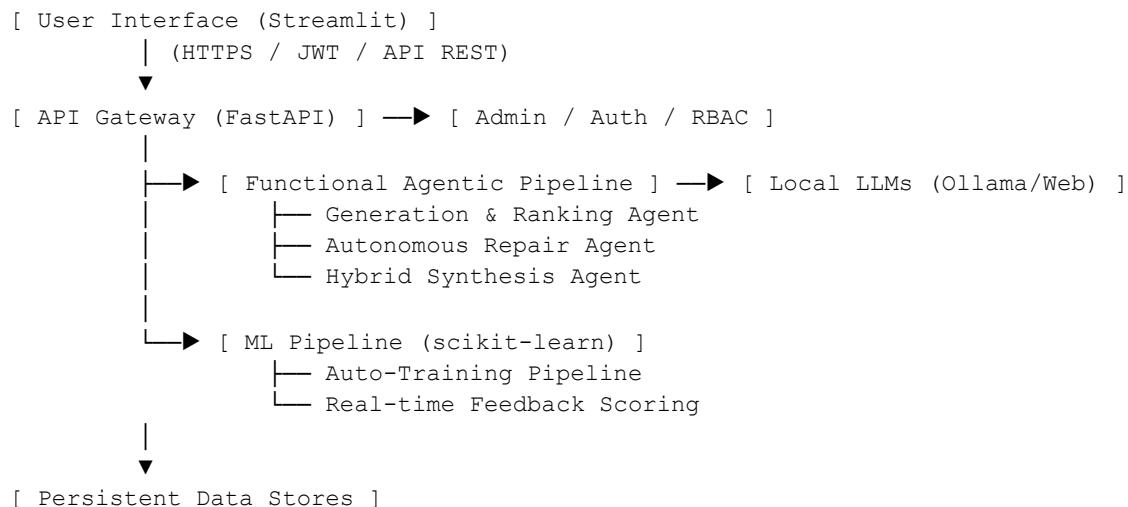
At its core, Sentinel leverages a decoupled client-server architecture. The backend operates robust, specialized Agentic Pipelines (combining prompt generation, SQL validation, autonomous error-recovery, and data synthesis) controlled by a high-concurrency FastAPI orchestrator. It provides three primary modalities of Retrieval-Augmented Generation (RAG): SQL RAG for structured transactional data, Graph RAG for interconnected entity analysis (tracking money laundering across accounts via Neo4j), and Multimodal RAG for unstructured evidence like documents, images, and audio utilizing FAISS vector indexing.

By integrating these capabilities with real-time Machine Learning pipelines (Random Forest, XGBoost) and persistent graph databases, Sentinel allows users to instantly transition from asking free-form natural language questions to deploying highly predictive fraud models.

## 2. Integrated Design

### 2.1 System Architecture & Data Flow

The architecture adheres to a strict decoupled pattern: a reactive Streamlit frontend communicates securely via JWT-authenticated REST APIs to a high-concurrency FastAPI backend. The backend acts as the central orchestrator, dynamically routing analytical requests through an ordered, functional Agentic Pipeline (SQL Generation, Candidate Ranking, Self-Recovery, and Reconciler agents), the supervised ML pipeline, or directly to the persistent data stores.



## Introduction to Fraud detection AI Workbench (version pro)

- SQLite (Raw structured data mappings)
- Neo4j (Entity graphs & taxonomic relations)
- FAISS (Vector embeddings for KB scaling)

### 2.2 UI Design Strategy

The UI architecture is entirely modularized under the `src/views/` directory. Each major feature (e.g., Data Hub, ML Workflow) is encapsulated in its own package, rendering state-aware tabs dynamically based on user context. The global application employs a 'Crimson Dark Mode' theme, utilizing conditional CSS injections to highlight critical alerts and interactive visual elements. Session State is rigorously managed to ensure that contextual data (such as a selected AI model, filter parameters, or a specific uploaded dataset) persists seamlessly as the user navigates fluidly between analytical tools.

## 3. Security Design

Security operates on a zero-trust model heavily focused on Role-Based Access Control (RBAC) and strict API perimeter defense.

### 3.1 Login Implementations & RBAC

Authentication uses standard OAuth2 architecture with JSON Web Tokens (JWT). Upon providing valid credentials, the backend (FastAPI) securely verifies password hashes utilizing BCrypt. It then issues a short-lived access token containing role scopes. The frontend caches this token locally and strictly attaches it to the header of every outbound API request.

RBAC is enforced dual-layer. On the frontend, the UI constructs the navigation sidebar by querying the `/admin/permissions` API endpoint, hiding pages the user cannot access. On the backend, routes enforce access internally using parameterized dependency injection (`RequiresRole`), terminating unauthorized requests before any business logic executes.

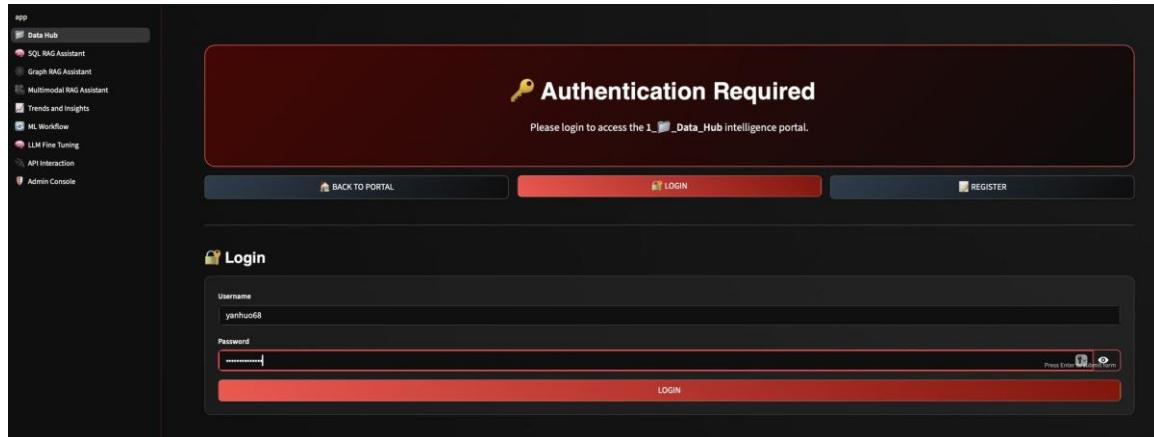
### 3.2 Data Security (SQL Anti-Injection)

The SQL RAG module employs an explicit Abstract Syntax Tree (AST) validator utilizing the `sqlglot` dependency. It inherently rejects LLM-hallucinated queries attempting destructive DROP, DELETE, UPDATE, INSERT, or ALTER commands. It further restricts queries to explicitly approved reporting tables, fully neutralizing prompt injection vulnerabilities.

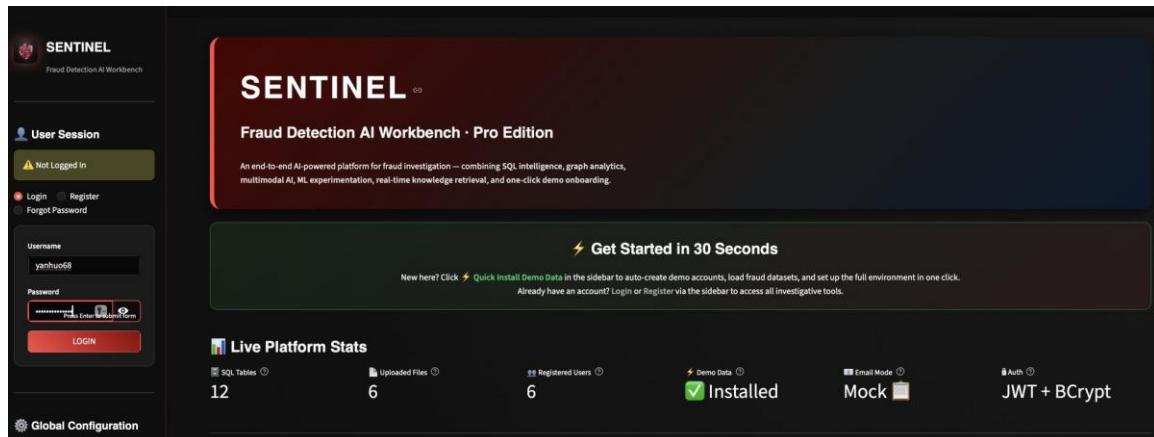
# Introduction to Fraud detection AI Workbench (version pro)

## 3.3 Demo

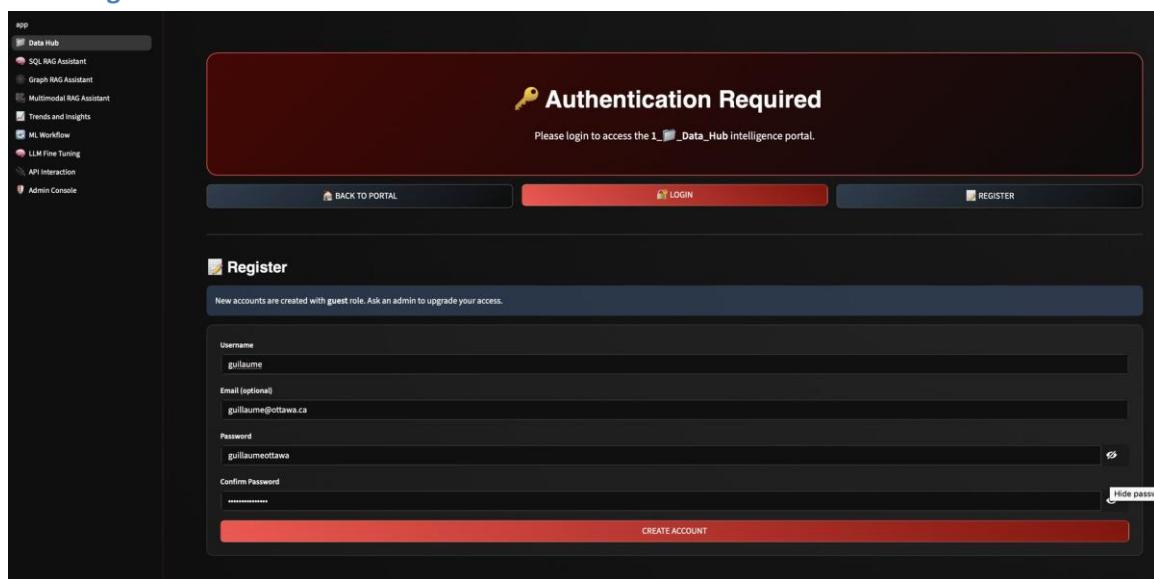
### 3.3.1.1 Login Way 1



### 3.3.1.2 Login Way 2



### 3.3.2 Register



## Introduction to Fraud detection AI Workbench (version pro)

### 3.3.3 Admin Console – User Management

The screenshot shows the 'User Management' section of the Admin Console. It lists three users:

ID	Username	Email	Role	Security	Delete
1	admin	[Set Email]	admin	guest	[Reset]
2	guest	[Set Email]	admin	data_scientist	[Reset]
3	scientist	[Set Email]	data_scientist		[Reset]

### 3.3.4 Admin Console - Role Management

The screenshot shows the 'Role Management' section of the Admin Console. It lists three roles:

- Role: guest
- Role: data\_scientist
- Role: admin

Under the 'Role: admin' section, it shows the 'Permissions (Allowed Pages)' for the admin role:

- 1. Data\_Hub
- 2. SQL\_RAG\_Assistant
- 3. Graph\_RAG\_Assistant
- 4. Multimodal\_RAG\_Assistant
- 5. Trends\_and\_Insights
- 6. ML\_Workflow
- 7. LLM\_Fine\_Tuning
- 10. API\_Interaction
- 11. Admin\_Console

A 'SAVE PERMISSIONS' button is located at the bottom right.

## 4. UI1 - Home

### 4.1 UI Design

The Home dashboard utilizes a responsive CSS grid layout mapping to three core investigative pillars: Intelligence, Visualization, and AI Core. It displays dynamic real-time system metrics (like active model count and Neo4j node connections) alongside action buttons governed by the authenticated user's RBAC role.

### 4.2 User Manual

Step 1: Review the system health on the central dashboard immediately upon login.

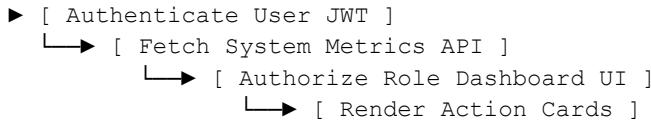
Step 2: If the system returns zero metrics, click the 'Quick Install Demo Data' fast-action

## Introduction to Fraud detection AI Workbench (version pro)

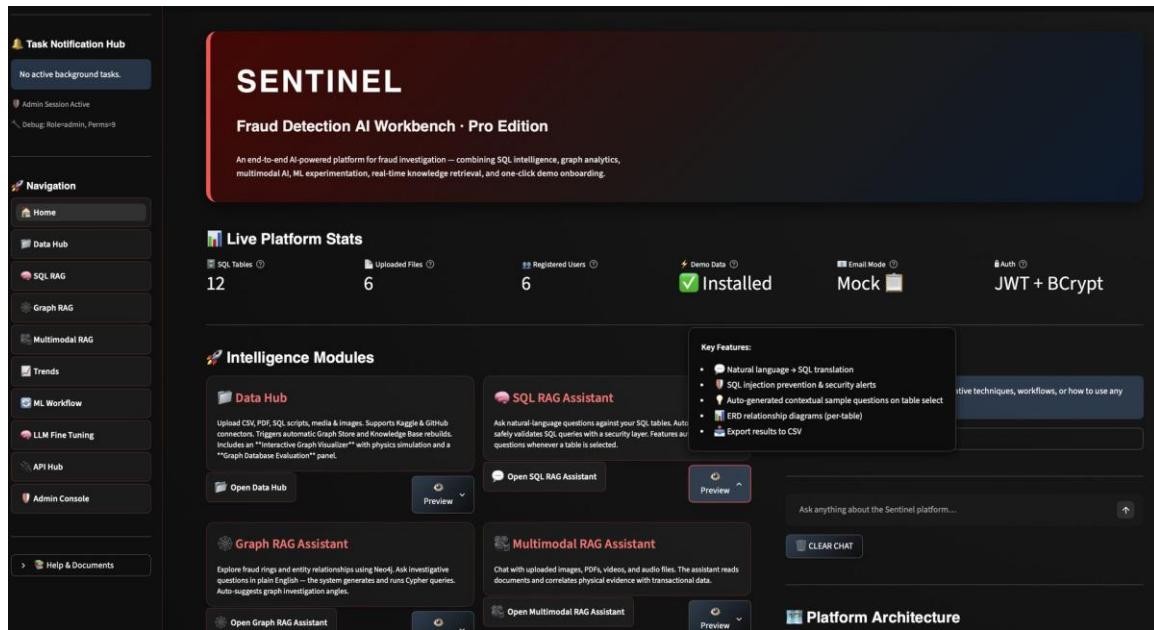
button to securely populate the SQLite databases and Neo4j graph.

Step 3: Click any primary module tile to bypass standard sidebar navigation and jump directly into an investigation.

### 4.3 Logic Workflow



### 4.4 Demo



When user initial comes to the system, above is home page. If user does not login, the page will redirect to Login page. If user has already logged in, user can click Preview to see key features or click Open xxx to go to desired page.

## 5. UI2 - Data Hub

### 5.1 UI Design

The Data Hub presents a tab-driven interface (Local Uploads, External Sources, Database Management) utilizing highly responsive file-uploader drop-zones and paginated dataframes. Crucially, it embeds a real-time progress bar array that visually indicates concurrent NLP embedding and graph ingest operations.

### 5.2 User Manual

Step 1: Open the 'Local Uploads' tab and drag a structural CSV file into the drop-zone.

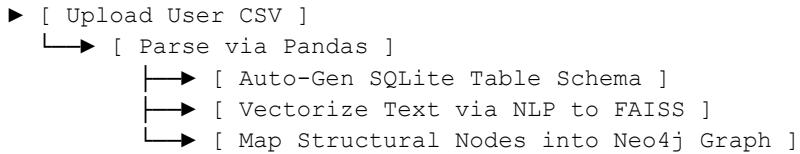
Step 2: Once uploaded, the backend auto-determines schema types. Click 'Rebuild'

## Introduction to Fraud detection AI Workbench (version pro)

Knowledge Base' to slice text content and embed it into FAISS.

Step 3: Map the CSV columns to Node/Edge identifiers to trigger a Neo4j ingestion pipeline.

### 5.3 Logic Workflow

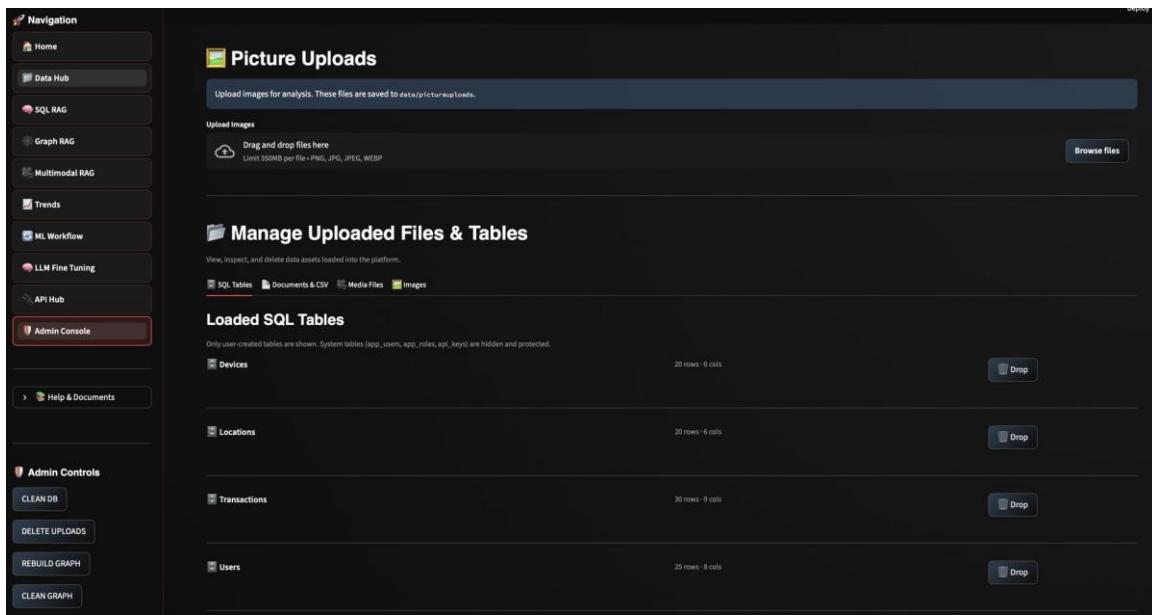


### 5.4 Demo

The screenshot shows the Fraud Detection AI Workbench Data Hub interface. On the left, there's a sidebar with sections for SENTINEL, User Session (logged in as admin), Global Configuration (API Keys, Active LLM set to openai:gpt-4-mini), Task Notification Hub (No active background tasks), and Admin Session Active (Debug: Role-admin, Perm-0). The main area is titled "Data Hub (via FastAPI backend)". It has four tabs: "Upload Files", "Run SQL Script", and "Media Uploads (Audio/Video)" are visible, while "External Data (Kaggle/GitHub)" is hidden. Under "Upload Files", there's a "Drag and drop file here" input field with a limit of 350MB for CSV, PDF, TXT, JSON, MD files. Under "Run SQL Script", there's a similar input field for SQL files. Under "Media Uploads", there's a note about uploading media files for multimodal RAG analysis. All sections have "Browse files" and "LOAD DEMO CSV" or "EXECUTE DEMO SQL" buttons.

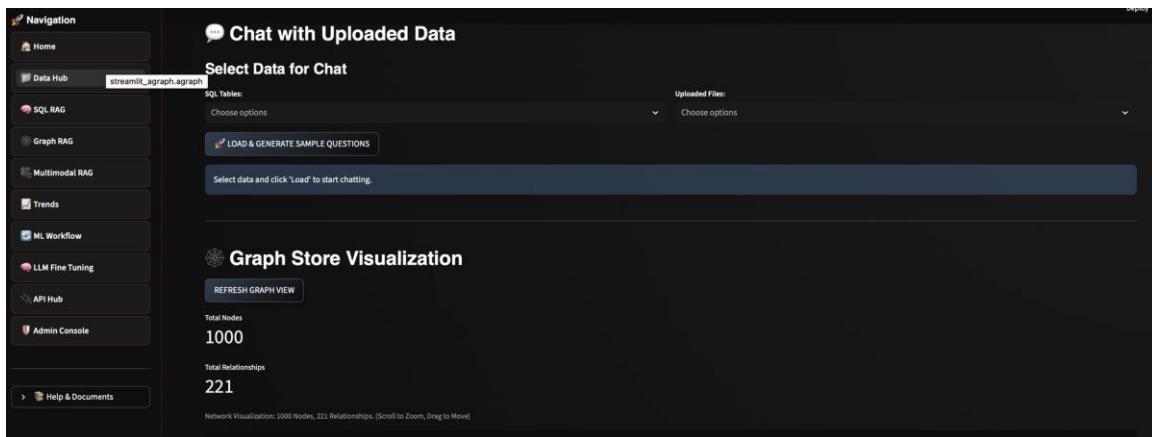
In the Data Hub, users can upload files (e.g. CSV/JSON/XML file, PDF file, media file (e.g. \*.mp3, \*.mp4, and so on), execute one or more SQL file(s). After uploading, for structure data can go to database, for unstructured data can go to vector store. After clicking Rebuild Graph button, all data no matter in database or vector store can be saved in graph store.

## Introduction to Fraud detection AI Workbench (version pro)



The screenshot shows the Data Hub section of the Fraud detection AI Workbench. The sidebar on the left contains links for Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main area is titled "Picture Uploads" and "Manage Uploaded Files & Tables". Under "Manage Uploaded Files & Tables", there is a section for "Loaded SQL Tables" which lists three tables: Devices, Locations, and Transactions. Each table has a "Drop" button next to it.

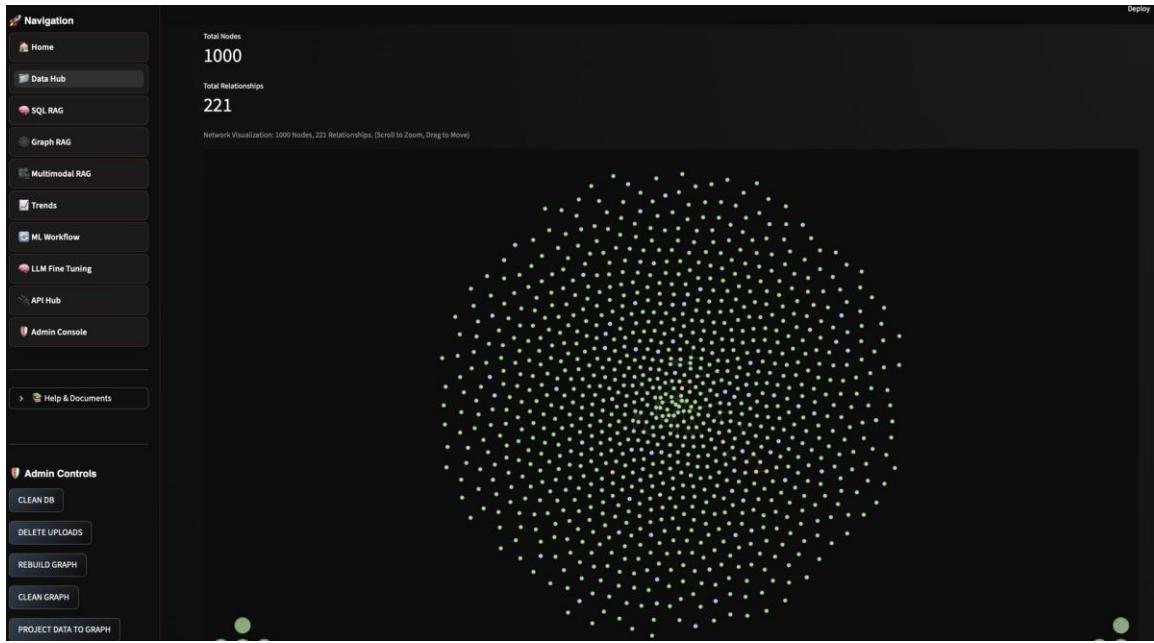
In the Data Hub, users can see the loaded tables and delete any one table.



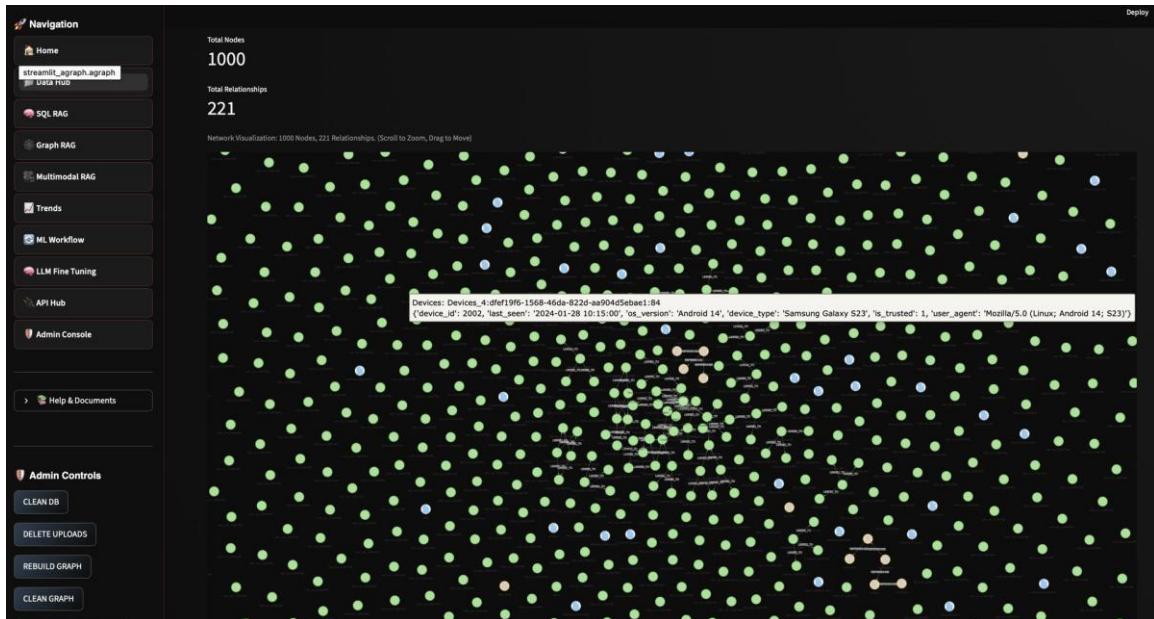
The screenshot shows the Chat with Uploaded Data section of the Fraud detection AI Workbench. The sidebar on the left contains links for Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main area is titled "Chat with Uploaded Data" and "Select Data for Chat". It includes a "Graph Store Visualization" section showing "Total Nodes: 1000" and "Total Relationships: 221".

After uploading, user can ask a question in the chat using natural language. If user does not feel satisfied to the answer, user can manually click "Rebuild Graph" button in the sidebar, then after rebuilding graph store, user can ask the same questions again.

## Introduction to Fraud detection AI Workbench (version pro)

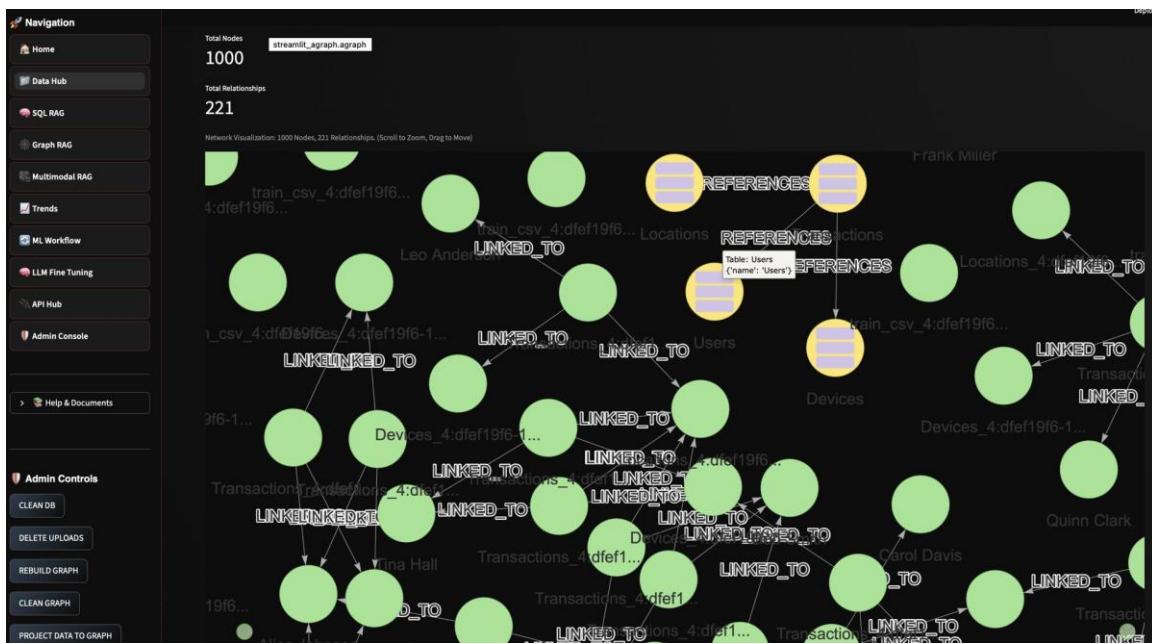


In Data Hub, users can see the whole graph and use zoom in/out and shift left/right to check the nodes.



If users click zoom in, user can find the relationship between nodes.

## Introduction to Fraud detection AI Workbench (version pro)



When user zooms deeper, user can see more detail of nodes.



If user clicks on one node, user can see the node detail under the graph visualization.

## Introduction to Fraud detection AI Workbench (version pro)

Label	Count
fraud_detection_dataset_mock_data.csv	51005
fraud_detection_dataset.csv	51000
train.csv	891
Document	68
OrderDetails	41
Orders	40
Transactions	30
Users	25
Locations	20
Devices	20
Customers	20
Table	15
app_users	6
api_keys	4
Products	2
Categories	2

From Graph database Evaluation, users can see the details of Infrastructure Health Check.

Query	Status	Latency (ms)	Result
0 Total Nodes	✓	1.55	cnt=103192
1 Total Relationships	✓	0.96	cnt=221
2 Avg Degree	✗	N/A	
3 Max Connections	✓	55.84	deg=23

From Graph database Evaluation, users can see Query Performance Profiling details.

## Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the Graph Database Evaluation page. The sidebar includes Home, Data Hub (selected), SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, Admin Console, Help & Documents, Admin Controls (CLEAN DB, DELETE UPLOADS, REBUILD GRAPH), and Deploy. The main content area has a title "Graph Database Evaluation" with a sub-instruction: "A deep diagnostic of your Graph Store – infrastructure health, query performance, data quality, and retrieval capability." A red button "RUN FULL EVALUATION" is at the top right. Below it, the "Overall Score: 85/100 (Grade B)" is shown with a note: "Runs ~10 Cypher diagnostics. Usually takes < 5 seconds." A "Issues & Recommendations" section lists "102992 orphaned nodes with no relationships detected." and "103071 nodes are missing name/title/id properties." Below this is a tab bar with Health, Performance, Quality (selected), and Retrieval. The "Graph Data Quality Metrics" section contains tables for Total Nodes (103192), Total Edges (221), Orphaned Nodes (102992 with a "Clean up" button), Duplicate Nodes (0 with a "Clean" button), Graph Density (0.000000), and Nodes Missing ID Prop (103071 with a "Add identifiers" button).

From Graph Database Evaluation, users can click third tab to see graph Data Quality Metrics.

The screenshot shows the Retrieval Quality Assessment page. The sidebar is the same as the previous one. The main content area shows an "Overall Score: 85/100 (Grade B)". Below it is the "Retrieval Quality Assessment" section. It displays three metrics: Depth-2 Reachable Pairs: 305, Avg Path Length: 1.275, and Traversal Latency: 27.27 ms. It also shows a "Keyword Search ('fraud'): 33 matching nodes found in 29.07 ms". Below this is a "Node Connectivity Coverage" section showing 0.4% coverage with a note: "Many isolated nodes". At the bottom, a note states: "Retrieval quality reflects how well the graph can support graph-based R&G queries. Higher coverage and lower latency = better R&G performance."

From Graph Database Evaluation, users can click fourth tab to see Retrieval Quality Assessment details.

## 6. UI3 - SQL RAG

### 6.1 UI Design

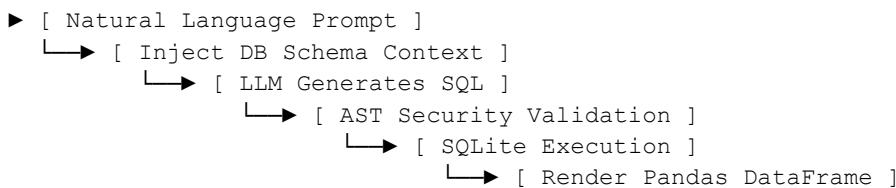
Constructed around a specialized split-screen IDE layout: the right-hand sidebar is rigidly fixed displaying dynamic active schema dictionaries and auto-generated sample questions. The primary left pane provides an endless-scroll conversational interface supporting dynamic rendering of Markdown, Pandas DataFrames, and Syntax-highlighted SQL blocks.

## Introduction to Fraud detection AI Workbench (version pro)

### 6.2 User Manual

- Step 1: Select an active target database using the header dropdown selector.
- Step 2: Type a natural language investigation metric (e.g., 'Return all accounts initiating over 5 transfers exceeding \$10k in 48 hours').
- Step 3: Review the AI's generated SQL, validate the query logic, and download the returned dataframe as an operational CSV.

### 6.3 Logic Workflow



### 6.4 Demo

The screenshot shows the Fraud Detection AI Workbench interface. On the left, there is a sidebar with the following sections:

- SENTINEL**: Shows the Fraud Detection AI Workbench logo.
- User Session**: Shows "Logged in as: admin" and a "LOGOUT" button.
- Global Configuration**: Includes "API Keys", "Select Active LLM" (set to "openai:gpt-4o-mini"), and "Active Model" (also set to "openai:gpt-4o-mini").
- Task Notification Hub**: Shows "No active background tasks."
- Admin Session Active**: Shows "Debug: Roleradmin, Perm:99".

The main content area is titled "SQL RAG Assistant (Multi-Agent)". It features a sidebar with "Suggested Questions" (Q1, Q2, Q3) and a "Default table for fallback (If JOIN fails)" dropdown set to "fraud\_detection\_dataset\_mock\_data\_csv". There is a "SUGGEST QUESTIONS" button. The main input area has a placeholder "Ask a question about your data (natural language):" followed by a text input field containing "e.g., What is the average transaction amount by type over the last 30 days?". Below the input field is a "RUN SQL + RAG PIPELINE" button and a note: "Start a new investigation by running the SQL pipeline above."

In SQL RAG page, users can input a question about structured data or click “Suggest Questions” button to generate sample questions by system. Users can click any one sample question; the sample question will automatically display in the input area. Then, users can click “RUN SQL + RAG PIPELINE” button to retrieve the results.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Investigative Findings' section of the AI Workbench. On the left is a navigation sidebar with various tabs like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main area has a title 'Investigative Findings' and a sub-section 'Winning Query'. It displays a SQL query:

```
sql
SELECT u.risk_score, COUNT(t.transaction_id) AS fraudulent_transactions
FROM Users u
LEFT JOIN Transactions t ON u.user_id = t.user_id AND t.is_fraudulent = 1 AND t.time >= datetime('now', '-1 month')
GROUP BY u.user_id;
```

Below this is a 'Data Preview' table with columns: risk\_score, fraudulent\_transactions, and fraud\_risk\_score. The data is as follows:

risk_score	fraudulent_transactions	fraud_risk_score
0	1.2	0
1	4.8	0
2	2.1	0
3	1	0
4	7.5	0
5	3.2	0
6	5.1	0
7	0.8	0
8	2.5	0
9	8.9	0

At the bottom is a 'SQL Analysis' section with a heading 'Explanation of the SQL Query and Its Context'.

In the result of SQL RAG PIPELINE, it displays the most candidate of SQL to use query to the database and display data preview of the table.

The screenshot shows the 'Analysis of SQL Queries' section of the AI Workbench. The left sidebar is identical to the previous screenshot. The main area has a title 'Analysis of SQL Queries' and a detailed explanation of three SQL candidates:

- Candidate 1:** This query attempts to calculate the average risk score of users who have made fraudulent transactions in the last month. However, it uses a `SUM` and filters for fraudulent transactions in the `WHERE` clause, which may lead to an empty result set if no users have made fraudulent transactions in the specified time frame.
- Candidate 2:** This query uses a `LEFT JOIN` to include all users, regardless of whether they have made fraudulent transactions. It counts the number of fraudulent transactions for each user but does not calculate the average risk score. The results show that all users have zero fraudulent transactions, which indicates that no users have committed fraud in the last month.
- Candidate 3:** Similar to Candidate 1, this query calculates the average risk score and counts fraudulent transactions. However, it uses a `SUM` with a `CASE` statement to count fraudulent transactions. Like Candidate 1, it may return an empty result set if no transactions meet the criteria.

Below this is a section for 'Explanation of Result Differences' and 'Correctness of Candidates'.

At the bottom is a 'Final SQL Query Suggestion' section with a heading 'To achieve a comprehensive analysis, the following SQL query can be used:' and a code block:

```
SELECT AVG(u.risk_score) AS average_risk_score,
       COUNT(t.transaction_id) AS fraudulent_transactions
  FROM Users u
 LEFT JOIN Transactions t ON u.user_id = t.user_id AND t.is_fraudulent = 1 AND t.time >= datetime('now', '-1 month')
 GROUP BY u.user_id;
```

This is a concrete analysis of SQL queries. Because LLM can generate at most 3 SQL queries and pick up the most semantic matched SQL query as the best candidate query to search data in the database.

## Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Hybrid Synthesis Report' section. On the left, there's a sidebar with 'Task Notification Hub' showing 'No active background tasks.' and a list of navigation items including Home, Data Hub, SQL RAG (which is selected), Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main content area has a title 'Analytic Report on User Risk Scores and Fraudulent Transactions' and a sub-section 'SQL Result Interpretation'. It includes a note about the SQL query results and a table:

risk_score	fraudulent_transactions
1.2	0
4.8	0
2.1	0
1	0
7.5	0
3.2	0
5.1	0
0.8	0
2.5	0
8.9	0
1.8	0
6.3	0
2	0
4.1	0
1.5	0
7.2	0
2.9	0
5.5	0
1.1	0
8.2	0

This is a description of hybrid synthesis analysis report for user review.

The screenshot shows the 'Fraud Risk Intelligence' section. The sidebar is identical to the previous one. The main content area has a title 'Fraud Risk Intelligence' and several sections: '1. Interpretation of the Fraud Risk Score' (explaining the score ranges from 0.0 to 1.0), '2. High-Risk Score Ranges' (listing Low Risk (0.0 - 0.3), Moderate Risk (0.3 - 0.7), and High Risk (> 0.7)), '3. Patterns Between Transaction Attributes and High Risk' (mentioning Fraudulent Transactions, Transaction Amount, and Transaction Type), '4. Suggestions for Improvement' (listing Machine Learning Model, Additional Features, and Real-time Monitoring), and a 'Report Export' section with a 'GENERATE PDF HUB' button.

In the last part, users can find Fraud risk intelligence analysis report.

## 7. UI4 - Graph RAG

### 7.1 UI Design

Mechanically mirrors the SQL RAG split-screen but substitutes relational schemas for intuitive Node/Edge taxonomy visualization. It integrates a physics-simulated PyVis network Javascript element directly via an interactive `st.components.v1.html` rendering block, situated immediately below the prompt intake.

## Introduction to Fraud detection AI Workbench (version pro)

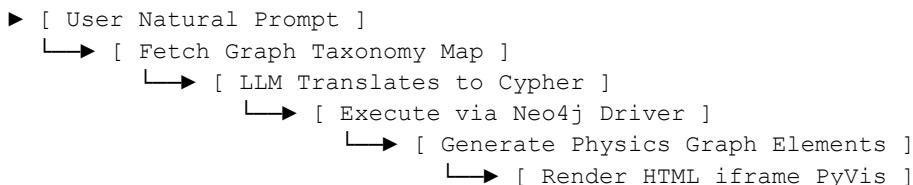
### 7.2 User Manual

Step 1: Choose an active Graph taxonomy matrix.

Step 2: Query the agent for overlapping attributes (e.g., 'Visualize the sub-graph involving IP Address 192.168.1.5').

Step 3: The interface retrieves the entity map. Manually drag nodes to rearrange the cluster layout, hover for deep property inspection, and review the AI-narrated summary of the identified fraud syndicate.

### 7.3 Logic Workflow



### 7.4 Demo

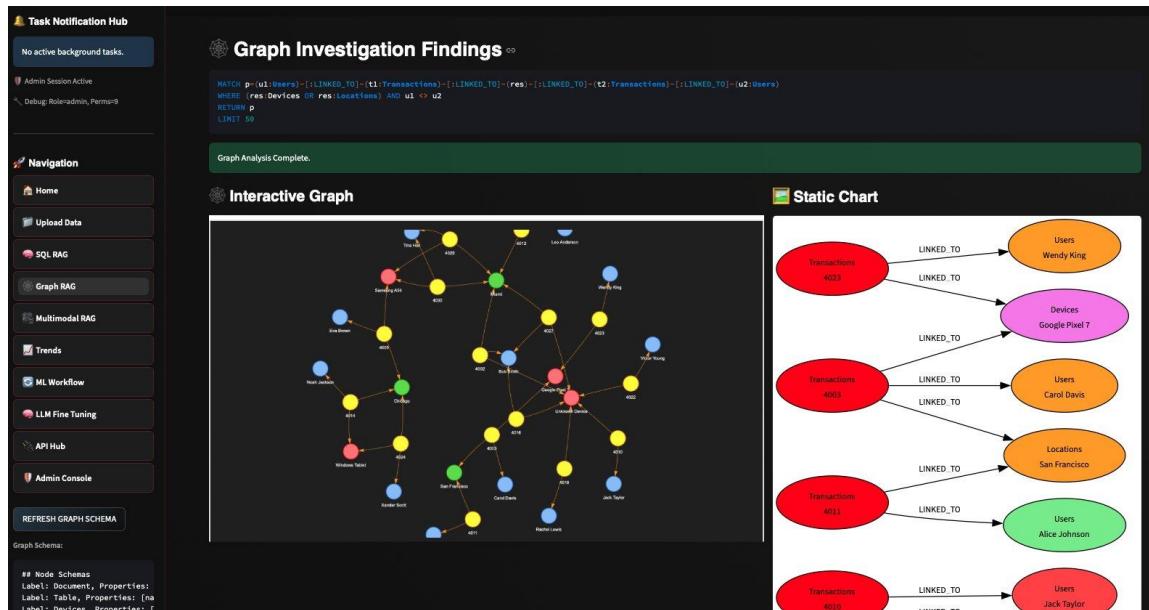
The screenshot shows the Graph RAG Assistant (Neo4j) interface. On the left, there's a sidebar with various navigation links: Home, Upload Data, SQL RAG, Graph RAG (which is highlighted), Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. Below the sidebar is a 'REFRESH GRAPH SCHEMA' button. The main content area is titled 'Graph RAG Assistant (Neo4j)'. It features a 'Suggested Questions' section with nine numbered boxes (Q1-Q9). Below that is a 'One-Click Analysis' section with three buttons: 'Fraud Rings', 'Key Influencers', and 'Risk Blast Radius'. Underneath is a 'Path Finder' section with two dropdown menus: 'Find connection between two Users' and 'Find Lookalikes (Similarity/Clones)'. At the bottom, there's a text input field asking 'Ask a question about relationships/network:' followed by a placeholder text: 'How do different locations cluster based on the number of transactions linked to them, and what are the paths to the users associated with those transactions?'. A 'RUN GRAPH QUERY' button is located at the very bottom of this section.

In **Graph RAG (Graph-based Retrieval-Augmented Generation)**, the system doesn't just retrieve similar chunks. It builds and reasons over a knowledge graph (entities + relationships). Users can have a few options to do graph RAG:

- Option 1: Users can input his/her own question.
- Option 2: Users can click "Suggest Graph Questions" to generate at most 9 questions then user can pick anyone question.
- Option 3: System provide one-click analysis which contains several buttons, user can click any one button
  - Fraud Rings

## Introduction to Fraud detection AI Workbench (version pro)

- Fraud rings = clusters of entities that collaborate in suspicious patterns.
- Path Finder
  - Path Finder identifies the relationship path between two entities in the graph.
- Key Influence
  - This identifies which node (entity) has the most influence in a network.



In the Graph Investigation, first, users can see the best candidate of graph query language for graph query. Then, user can see Interactive Graph on the left side and static chart analysis on the right side.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the Fraud Analysis Report page. On the left is a navigation sidebar with options like Home, Upload Data, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. Below the sidebar is a 'REFRESH GRAPH SCHEMA' button and a 'Graph Schema' section. The main content area starts with an 'AI Summary' section featuring a small diagram of a red node connected to an orange 'Users' node labeled 'Noah Jackson' with the edge label 'LINKED\_TO'. Below this is the 'Fraud Analysis Report' section with a 'Findings' heading. It states: 'The analysis of the relationship graph revealed several indirect connections between users and transactions, indicating potential fraudulent activity. The following key points were noted:'. A numbered list follows:

1. User Connections:
  - o Multiple users (Jack Taylor, Bob Smith, Victor Young, Rachel Lewis, etc.) are linked through transactions, suggesting possible collusion or shared fraudulent behavior.
  - o Bob Smith appears frequently as a central node, connecting with various users, which may indicate he is a pivotal figure in these transactions.
2. User Risk Scores:
  - o Users have varying risk scores, with Jack Taylor (8.3) and Victor Young (6.8) being among the higher scores, indicating a greater likelihood of fraudulent activity.
  - o Bob Smith, despite having a lower risk score (4.8), is under review, which may suggest previous suspicious activity.
3. Account Status:
  - o Several users are marked as "suspended" or "under review," indicating ongoing investigations or previous issues that warrant further scrutiny.
  - o Users with lower risk scores but under review status may still pose a risk due to their connections with higher-risk users.
4. Device and Location Links:
  - o The paths include connections through devices and locations, which could be used to further investigate the nature of these transactions and the legitimacy of the users involved.

Below the findings is a 'Risk Assessment' section with an 'Overall Risk Score: 7.5/10' and a list of risk levels:

- High Risk: Users with risk scores above 7 (e.g., Jack Taylor, Victor Young) are considered high risk and warrant immediate attention.
- Moderate Risk: Users with scores between 4 and 7 (e.g., Bob Smith, Rachel Lewis) should be monitored closely.
- Low Risk: Users with scores below 4 are currently not flagged but should be observed for any changes in behavior or connections.

After graph query, users can see AI insights of fraud analysis.

The screenshot shows the Recommended Actions page. On the left is a navigation sidebar with options like User Session (Logged in as: admin, Role: admin, LOGOUT), Global Configuration (API Keys, Select Active LLM: openai:gpt-4o-mini), Task Notification Hub (No active background tasks, Admin Session Active, Debug: Role=admin, Perms=9), and a GENERATE PDF REPORT button. The main content area starts with a 'Recommended Actions' section. It lists six categories with sub-points:

1. Immediate Investigation:
  - o Conduct a detailed investigation into the transactions linked to high-risk users, particularly focusing on Jack Taylor and Victor Young.
  - o Review the transaction history of Bob Smith to understand his connections and any patterns of behavior that may indicate collusion.
2. Enhanced Monitoring:
  - o Implement enhanced monitoring for users with risk scores between 4 and 7, especially those under review.
  - o Set alerts for any transactions involving these users, particularly if they are linked to high-risk accounts.
3. User Interviews:
  - o Consider interviewing users with high-risk scores to gather more information about their activities and connections.
  - o Assess their understanding of the platform's policies and any potential unintentional violations.
4. Device and Location Analysis:
  - o Analyze the devices and locations associated with these transactions to identify any patterns or anomalies that could indicate fraudulent activity.
  - o Cross-reference device IDs and locations with known fraudulent activities to identify potential links.
5. Policy Review:
  - o Review current policies regarding user verification and transaction approvals to strengthen defenses against potential fraud.
  - o Consider implementing stricter verification processes for high-risk users and transactions.
6. Training and Awareness:
  - o Provide training for staff on recognizing signs of fraud and the importance of reporting suspicious activities.
  - o Increase awareness among users about the risks of sharing accounts or devices, which can lead to fraudulent behavior.

By taking these actions, the organization can mitigate risks associated with fraudulent activities and enhance the overall security of its operations.

Users can see Recommended Actions.

## Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Fraud Analysis Report: Key Influencers' page. The sidebar on the left includes sections for Task Notification Hub, Admin Session Active, Navigation (Home, Upload Data, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub), and Admin Console. A 'REFRESH GRAPH SCHEMA' button is also present. The main content area displays findings based on relationship graph analysis, listing the top 10 users and devices with the highest degree centrality. A summary notes that Alice Johnson is the most influential user.

Users can see top impacts to key influencers for fraud analysis report.

## 8. UI5 - Multimodal RAG

### 8.1 UI Design

Segmented cleanly via inner tabs dividing logical intake flows: Audio, Image, and Document ingestion. The interface incorporates built-in HTML5 media players enabling contextual review, alongside real-time transcription loading spinners.

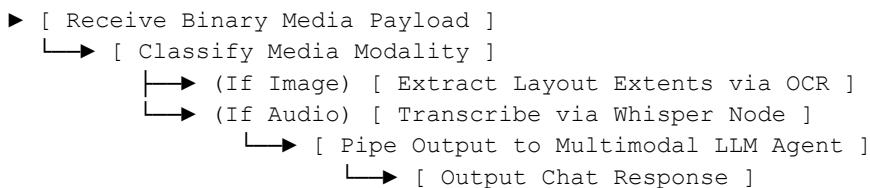
### 8.2 User Manual

Step 1: Upload a piece of localized evidence, such as a scanned invoice image or a suspicious WAV audio recording.

Step 2: Once uploaded, wait for the OCR or Whisper transcription block to finalize.

Step 3: Interact specifically with that media via the chat element, instructing the AI to 'Compare the extracted routing number to standard banking indices'.

### 8.3 Logic Workflow



# Introduction to Fraud detection AI Workbench (version pro)

## 8.4 Demo

The screenshot shows the Multimodal RAG Assistant interface. On the left, there's a sidebar with sections like 'SENTINEL', 'User Session' (logged in as admin), 'Global Configuration' (Active Model: openai-gpt-4o-mini), and 'Task Notification Hub' (No active background tasks). The main area has tabs for 'Multimodal RAG Assistant', 'Picture RAG', 'Text RAG', 'Web RAG', and 'Aware RAG'. Under 'Multimodal RAG Assistant', it says 'Select file [av]: PEI lighthouse at the risk.mp3'. Below this is a 'Preview' section with a play button and a timestamp of 02:07. To the right is an 'Actions' button labeled 'TRANSCRIBE & ANALYZE'. Further down is a 'Content Description' section with a transcript of a speech about a lighthouse at risk due to coastal erosion. At the bottom is a 'DOWNLOAD TEXT (TXT)' button. On the right side, there's an 'AI Insights' section with a summary and a 'GENERATE SUMMARY' button, containing a detailed text about the challenges of preserving a historic lighthouse.

In Multimodal RAG, users can extract information from audio or video In above sample, user get content description from a mp3 file and can generate summary or meeting minutes.

The screenshot shows the Multimodal RAG Assistant interface with a different sidebar. The sidebar includes 'Navigation' (Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, Admin Console) and 'Task Notification Hub' (No active background tasks). The main area shows the 'Multimodal RAG Assistant' tab selected. It displays an 'ML Project flowchart by' with 10 numbered steps: 1. Data Acquisition, 2. Data Cleaning & Labelling, 3. Exploratory Data Analysis, 4. Data Preprocessing, 5. Split Dataset, 6. Model Training, 7. Model Performance Evaluation, 8. Hyperparameter Tuning, 9. Final Evaluation, and 10. Model Deployment. There are also 'Train Set', 'Validation Set', 'Test Set', and 'Evaluator Model' components. An 'Actions' button labeled 'DESCRIBE & ANALYZE' is present. The flowchart illustrates the machine learning pipeline from data acquisition to deployment.

# Introduction to Fraud detection AI Workbench (version pro)

Based on a picture of workflow, user can use “Describe & Analyze” to get below information.

The screenshot shows the AI Workbench interface with a sidebar on the left containing various navigation options like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main area has a "ML Project Rowchart by" button at the top right. Below it, there are two main sections: "Content Description" and "AI Insights".

**Content Description:** This section includes a "Full Text" link and a "DOWNLOAD TEXT (.TXT)" button. It also contains a "Flowchart Details" section with numbered steps:

1. \*\*Problem Formulation\*\*  
- First step, placed at the top left.
2. \*\*Data Acquisition\*\*  
- Connected directly below Problem Formulation.
3. \*\*Data Cleaning & Labelling\*\*  
- Follows Data Acquisition in sequence.
4. \*\*Exploratory Data Analysis\*\*  
- Proceeds from Data Cleaning & Labelling.

**AI Insights:** This section includes a "SUMMARY" and "Meeting Minutes" tab, with "GENERATE SUMMARY" and "GENERATE PPT" buttons. It contains sections for "Main Topic", "Key Arguments", "Overall Conclusion", and "Presentation".

Based on detecting source language, user can translate to English and French. Users can generate PPT from summary or minute content.

The screenshot shows the AI Workbench interface with a sidebar on the left containing various navigation options like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning (which is highlighted in red), API Hub, and Admin Console. The main area has a "Chat with Content" section where users can ask questions. A history of questions and answers is shown:

- What the topic talk about?
- The topic discusses the process of a machine learning (ML) project, detailing the steps from problem formulation to documentation, including data acquisition, cleaning, analysis, model training, evaluation, tuning, deployment, and final documentation.
- Please tell me more detail step by step
- I don't find that information in the transcript.
- Please tell me the work flow
- The workflow of the ML project is as follows:
  1. Problem Formulation
  2. Data Acquisition
  3. Data Cleaning & Labelling
  4. Exploratory Data Analysis
  5. Data Preprocessing
  6. Split Dataset (into Train Set and Test Set)
  7. Train Set leads to Model Training
  8. Model Training connects to Model Performance Evaluation
  9. Validation Set connects to both Model Training and Model Performance Evaluation
  10. Model Performance Evaluation leads to Hyperparameter Tuning
  11. Hyperparameter Tuning connects to Final Evaluation
  12. Final Evaluation connects from Test Set (marked as Final Testing)
  13. Model Deployment follows Final Evaluation
  14. Documentation is the final step after Model Deployment.

Based on graph data for multimodal, users can chat to get I information.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the Multimodal RAG Assistant interface. On the left, there's a navigation sidebar with various tabs like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG (which is selected), Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main area has a header "Multimodal RAG Assistant" with a sub-header "Web Content RAG". It says "Upload files in Data Hub tab. Here you can transcribe and analyze them." Below that are buttons for "Audio/Video RAG", "Picture RAG", "Text RAG", "Web RAG", and "Aware RAG". A "Web URL" input field contains a Canadian news article URL, and a "PROCESS URL" button is shown. A green bar indicates "Web content extracted and indexed!". Below this is a link to "View Scrapped Content". To the right, there are sections for "Sentiment & Workflow" (with "ANALYZE SENTIMENT" and "GENERATE WORKFLOW" buttons) and "Translation & Docs" (with "AUTO-TRANSLATE" and "PRESENTATION" buttons). At the bottom, there's a "Chat with Web Content" section.

Based on Web content RAG, users can get sentiment analysis and draw the workflow shown below.

This screenshot shows the "Sentiment & Workflow" section of the Multimodal RAG Assistant. The left sidebar is identical to the previous screenshot. The main area has a "GENERATE WORKFLOW" button, which is highlighted in a green bar with the message "Diagram Created!". Below this is a complex workflow diagram. The diagram starts with a central node labeled "Work" and branches into several paths. One path leads to "Overall sentiment: Neutral". Another path leads to "Emotional Tone: Concerned yet Hopeful". A third path leads to "Key Emotional Drivers", which further branches into "Challenges and Anxiety", "Resilience and Optimism", and "Imposter Syndrome and Self-doubt". From "Challenges and Anxiety", it leads to "Hidden Implications" and "AI Strategic Insight". "Hidden Implications" leads to "Recommended Actions" like "Policy Development", "Community Engagement", and "Career Support". "AI Strategic Insight" leads to "Hidden implications" and "AI Strategic Insight". The "Key Emotional Drivers" path also leads to "AI Strategic Insight". The "Emotional Tone" path leads to "Overall sentiment" and "AI Strategic Insight". The "Overall sentiment" path leads to "Detected Language: English" and "Translation". The "Translation" section shows French text from a news article about young Canadians and their challenges, with options to "Gérer le service" or "Déconnexion". The "AI Strategic Insight" section shows a large block of French text from a student named Violet Rode, discussing her anxiety and the COVID-19 pandemic, along with "Business Wire" and "GlobeNewswire" links. The "Overall sentiment" path also leads to a "Download Diagram" button.

Based on extract Web information and source language, system can automatically translate to English and French.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Aware RAG' section of the AI Workbench. On the left, a sidebar navigation includes 'Home', 'Data Hub', 'SQL RAG', 'Graph RAG', 'Multimodal RAG', 'Trends', 'ML Workflow', 'LLM Fine Tuning', 'API Hub', and 'Admin Console'. The main area displays analysis results for a file named 'PEI lighthouse at the risk.mp3'. It shows a summary of 'Main Speakers' (Shia Dajabali and Scott McEwan), 'Dialogue Patterns', 'Key Contributions' (Shia Dajabali provides an overview, Scott McEwan expresses concern), and a note that there is no response needed as it's a report. At the bottom, there are tabs for 'Speaker-Aware', 'Time-Aware', 'Sentiment-Aware', and 'Confidence-Aware'.

In Aware RAG, there are 4 options for users:

- Speaker Aware
- Time Aware
- Sentiment Aware
- Confidence Aware

In Speak Aware, based on multimodal content, users can analyze speaker's purpose, dialogue patterns, key contributes, response, so on.

The screenshot shows the 'Multimodal RAG Assistant' section of the AI Workbench. The sidebar is identical to the previous screenshot. The main area shows the same analysis for 'PEI lighthouse at the risk.mp3' but focuses on 'Time-Aware Analysis'. It identifies a timeline from 1865 to 2022, noting the original building, coastal erosion, and the 2022 storm Fiona. It also identifies a sequence of events including the lighthouse's construction, its threats over time, and current protection efforts. Like the previous section, it includes tabs for 'Speaker-Aware', 'Time-Aware', 'Sentiment-Aware', and 'Confidence-Aware'.

In Time-aware analysis, users can analyze multimodal timeline and sequence of events.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the Fraud detection AI Workbench interface. On the left is a navigation sidebar with various options like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main area is titled "Aware RAG" and shows a file named "PEI lighthouse at the risk.mp3" selected for analysis. Below it, under "Sentiment-Aware Analysis", there's a summary: "Overall Sentiment: Negative", "Emotional Tone: Concerned, Urgent, Reflective", and "Key Emotional Drivers". A detailed list of findings follows, including concerns about coastal erosion, urgency of protection, and reflection on climate change. At the bottom, there are "AI Strategic Insight" recommendations.

In Sentiment-Aware analysis, users can analyze sentiment of multimodal. For example, you can see overall sentiment active or negative, emotional tone, key emotional drivers and insights.

The screenshot shows the Fraud detection AI Workbench interface. The navigation sidebar is identical to the previous screenshot. The main area is titled "Aware RAG" and shows the same file "PEI lighthouse at the risk.mp3" selected for analysis. Below it, under "Confidence & Quality Control", there's a summary: "Content Clarity: 7/10", "Information Density: Medium", "Ambiguity Check", "Source Reliability", and "Coverage Quality". A detailed list of findings follows, providing specific feedback on the narrative flow, repetition, and source reliability.

In Confidence Aware, users can analyze content quality, for example, content clarity, density, ambiguity check, source reliability and coverage quality, so on.

# Introduction to Fraud detection AI Workbench (version pro)

## 9. UI6 - Trends and Insights

### 9.1 UI Design

Serves as the primary BI reporting dashboard. Leverages visually reactive Plotly Express charts scaling automatically to container width. Employs advanced cascading drop-downs for dimensional X/Y axis parameterization, dynamic color grouping, and visual-type toggles (scatter, bar, histogram).

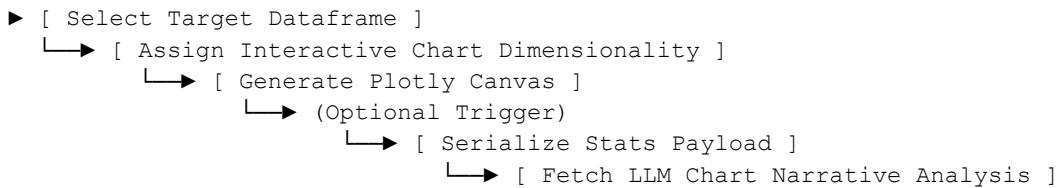
### 9.2 User Manual

Step 1: Select an ingested target dataset.

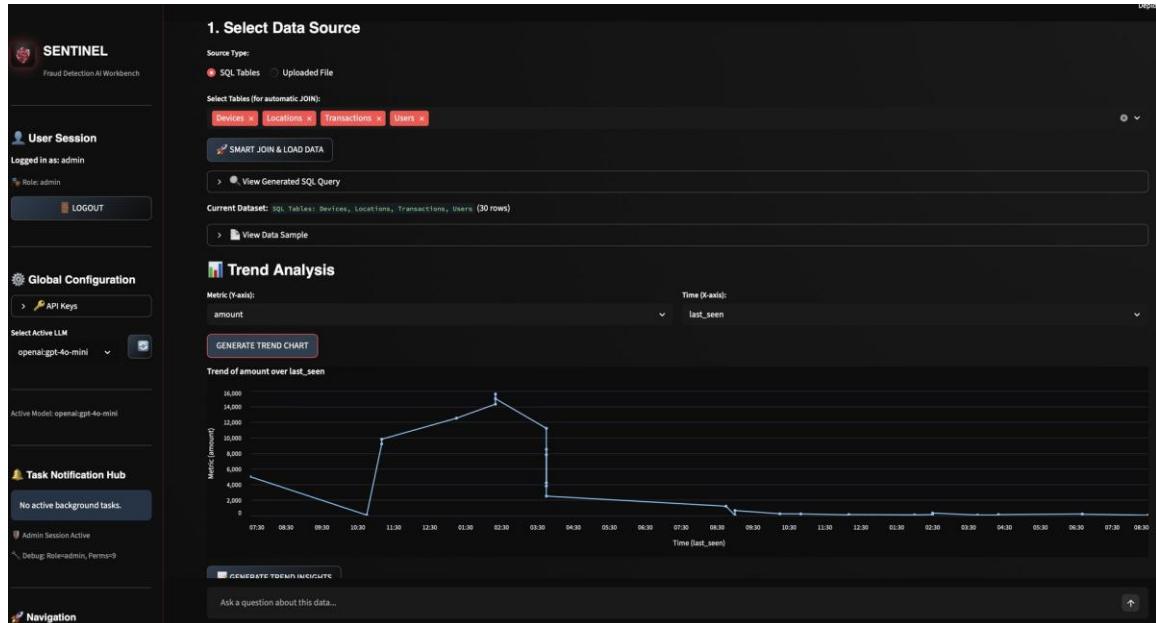
Step 2: Assign mathematical parameters to the X/Y axes to identify macro-trends visually (e.g., fraudulent volume plotted across transaction hours).

Step 3: Click 'Generate AI Insights'. The platform passes the statistical density bounds to the LLM to output a human-friendly narrative explaining the chart.

### 9.3 Logic Workflow



## 9.4 Demo



In Trends page, for multiple selected relational tables, after clicking "Smart join & data load" button, user can get Trend Analysis.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the Fraud Detection AI Workbench interface. On the left, there's a sidebar with sections for User Session (Logged in as: admin, Role: admin, Logout), Global Configuration (API Keys, Select Active LLM: openai/gpt-4o-mini), Task Notification Hub (No active background tasks), and Navigation (Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, Admin Console). The main content area is titled "Automated EDA" and contains a "GENERATE EDA REPORT" button. Below it is a "Numeric Stats" table:

	count	mean	std	min	25%	50%	75%	max
location_id	30	3007.9333	5.8659	3001	3003.25	3006.5	3011.75	3020
transaction_id	30	4015.5	8.8034	4001	4008.25	4015.5	4022.75	4030
user_id	30	111.4667	8.0675	101	103.5	111	118.75	125
device_id_1	30	2009.1	5.5792	2001	2005.25	2007	2013.5	2020
location_id_1	30	3007.9333	5.8659	3001	3003.25	3006.5	3011.75	3020
amount	30	4109.73	5340.1079	45.75	125.125	485	8325	15600
is_fraudulent	30	0.4333	0.504	0	0	0	1	1
user_id_1	30	111.4667	8.0675	101	103.5	111	118.75	125
account_age	30	256.1667	206.8136	15	92.5	195	365	720
risk_score	30	3.6133	2.4959	0.8	1.5	2.7	5.625	8.9

Below the table is a "AI Insights" section titled "EDA Summary for Fraud Analytics". It includes a "1. Distribution Characteristics" section with "Numeric Columns" for "Amount" and "Account Age". The "Amount" distribution is described as having a positive skew (mean > median) with outliers. The "Account Age" distribution is described as having a range from 15 to 720 days.

Based on relational data, users can get EDA analysis. EDA = Exploratory Data Analysis which is deeply explore data before building AI/ML models. EDA helps you:

- Discover patterns
- Detect outliers
- Understand feature relationships
- Validate assumptions
- Improve feature engineering

The screenshot shows the Fraud Detection AI Workbench interface. The sidebar is identical to the previous screenshot. The main content area has two sections: "Top Anomalies Detected" and "AI Anomaly Analysis".

**Top Anomalies Detected:** This section shows a table of anomalies:

device_id	device_type	os_version	last_seen	is_trusted	user_agent	location_id	city	state	country	risk_level	timezone	transaction_id	user_id	device_id_1	location_id_1	amount	type	
18	2019	Android TV Box	Android 11	2024-01-28 20:30:00	0	Mozilla/5.0 (Linux; Android 11; TV)	3012	London	England	UK	medium	None	4019	119	2019	3012	55	Online Purchase
23	2014	Windows Tablet	Windows 11	2024-01-27 23:10:00	0	Mozilla/5.0 (Windows NT 10.0; Win64; x64)	3003	Chicago	IL	USA	high	None	4024	124	2014	3003	9200	ATM Withdrawal

**AI Anomaly Analysis:** This section contains a descriptive text about the detected anomalies:

The detected anomalies share several common patterns: both transactions involve devices that are not typically associated with high-risk activities (an Android TV Box and a Windows Tablet), and both transactions occurred in major urban locations (London and Chicago). The first transaction is an online purchase with a low amount, while the second is a high-value ATM withdrawal. The user associated with the high-value transaction has a relatively short account age (190 days) and is under review, indicating potential risk. The combination of a high-risk device type, unusual transaction amounts, and the users' account statuses suggest these transactions deviate from typical behavior, raising red flags for potential fraud. The lack of trust in the devices and the high-risk nature of the transactions further contribute to their classification as anomalies.

**Geospatial Analysis (Top Locations):** This section features a bar chart titled "Top 18 by Transaction Volume" showing the top locations:

city	volume
Miami	1800
New York	1500
Boston	1200
San Francisco	1000
Sydney	800
Seattle	700
Berlin	600
London	500

Users can see top anomalies detected analysis and based on location column, make the Geospatial analysis for the top locations.

## Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the Fraud detection AI Workbench interface. On the left is a sidebar with various sections: Task Notification Hub (No active background tasks), Admin Session Active (Debug: Role:admin, Perms:9), Navigation (Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, Admin Console), and Admin Console. The main area is titled "Chat with Data". It features a question input field with placeholder "Which city has the highest transaction amount?", a response "Miami", and another question "What is the fraud rate by device type?". Below these are four buttons: "WHAT IS THE FRAUD RATE BY DEVICE TYPE?", "WHICH CITY HAS THE HIGHEST TRANSACTION AMOUNT?", "HOW MANY TRUSTED DEVICES ARE IN EACH STATE?", and "WHAT IS THE AVERAGE RISK SCORE BY USER ACCOUNT AGE?". A table titled "device\_type" is displayed, showing a list of devices and their corresponding fraud rates. At the bottom is a text input field "Ask a question about this data..." and a file upload icon.

device_type	fraud_rate
0 Android TV Box	0
1 Android Tablet	0
2 ChromeBook	0
3 Google Pixel 7	0
4 Linux Desktop	0
5 MacBook Pro	0
6 Samsung A54	1
7 Samsung Galaxy S23	0
8 Unknown Device	1
9 Windows Laptop	0

Users can use natural language to chat with data or use system generated sample question to ask questions.

## 10. UI7 - ML Workflow

### 10.1 Functionality Design

Functionally designed as an automated, multi-tiered pipeline leveraging scikit-learn and imbalanced-learn. The system decomposes the complexity of traditional data science workflows into sequential state-managed modules. It actively mitigates fraud datasets' notorious class imbalance using SMOTE (Synthetic Minority Over-sampling Technique) before executing model training. Furthermore, game-theoretic SHAP (SHapley Additive exPlanations) values are automatically computed to provide strict regulatory transparency into exactly why the algorithmic decision tree flagged a specific transaction.

### 10.2 UI Design

The interface spans multiple horizontal tabs (Build, Train, Tune, Score, Deploy, Monitor) to gracefully orchestrate the end-to-end Machine Learning lifecycle without overwhelming the analyst. Parameters and hyperparameter tuning fields are housed in expandable accordions. Outputs include visually striking KPIs (Accuracy, F1-Score) and interactive matplotlib/plotly charts for SHAP dependency curves and Confusion Matrices.

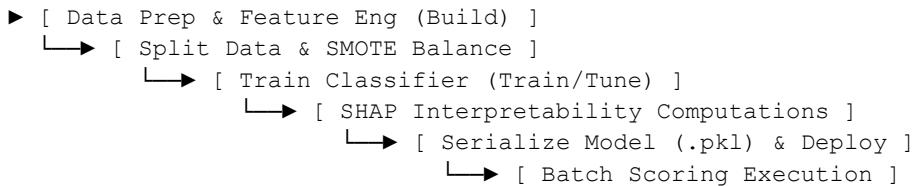
### 10.3 User Manual

Step 1: In the 'Build Pipeline' tab, select a dataset, assign target variables, and configure automated feature engineering (imputation, categorical encoding, SMOTE for imbalance).  
Step 2: In the 'Train Model' tab, choose an algorithm (e.g., Random Forest, XGBoost), configure the Train/Test split, and fit the model to view KPIs and SHAP feature importance.  
Step 3: Move to 'Tune Model' for GridSearchCV hyperparameter tuning.

## Introduction to Fraud detection AI Workbench (version pro)

Step 4: Use 'Batch Scoring' to apply your serialized model to new datasets, and 'Monitor' to observe data drift and performance decay over time.

### 10.4 Logic Workflow



### 10.5 Demo

The screenshot shows the 'ML Workflow' section of the AI Workbench interface. On the left is a sidebar with various navigation options like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow (which is selected), LLM Fine Tuning, API Hub, and Admin Console. The main area is titled 'End-to-End ML Workflow' and describes a unified workflow from building datasets to monitoring models. Step 1: Build Training Dataset is currently active. It has two sub-steps: 1.1 Select Source Data and 1.2 Data Preview. In 1.1, there's a file upload section for 'Fraud Detection Dataset.csv'. In 1.2, a table shows a preview of the loaded data with 5 rows and 13 columns. The columns include Transaction\_ID, User\_ID, Transaction\_Amount, Transaction\_Type, Time\_of\_Transaction, Device\_Used, Location, Previous\_Fraudulent\_Transactions, Account\_Age, Number\_of\_Transactions\_Last\_24H, Payment\_Method, and Fraudulent.

Transaction_ID	User_ID	Transaction_Amount	Transaction_Type	Time_of_Transaction	Device_Used	Location	Previous_Fraudulent_Transactions	Account_Age	Number_of_Transactions_Last_24H	Payment_Method	Fraudulent
0   T1	4174	1292.76	ATM Withdrawal	16	Tablet	San Francisco	0	119	13	Debit Card	0
1   T2	4507	1554.58	ATM Withdrawal	13	Mobile	New York	4	79	3	Credit Card	0
2   T3	1860	2395.02	ATM Withdrawal	None	Mobile	None	3	115	9	None	0
3   T4	2394	100.1	Bill Payment	15	Desktop	Chicago	4	3	4	UPI	0
4   T5	2130	1490.5	POS Payment	19	Mobile	San Francisco	2	57	7	Credit Card	0

In the end-to-end ML workflow, there are 6 steps to implement in current system:

- Build dataset
- Train model
- Live Scoring
- Model fine tuning
- Deploy Model
- Monitor Model

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Configuration' section of the AI Workbench. On the left, there's a sidebar with navigation links like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, and ML Workflow. The main area is titled '1.3 Dataset Health Check' with a checkbox 'Show Health Check' checked. Below it is a 'Missing Values' section with a warning message: '⚠ Missing values detected in 5 columns.' A table lists the missing counts for five columns: Transaction\_Amount (2520), Time\_of\_Transaction (2552), Device\_Used (2473), Location (2547), and Payment\_Method (2469). At the bottom is a button 'ANALYZE MISSING DATA RISKS'.

Before train data, users need to select target column and exclude columns. The user can do dataset health check.

This screenshot shows a more detailed view of the 'Missing Values' analysis. It includes a summary table of missing counts and a 'Missing Values Overview' section with a note about choosing appropriate imputation strategies based on data nature and context. Below is a 'Imputation Strategies' section with three numbered points corresponding to the columns:

- 1. Transaction\_Amount (2520 missing values)
  - Mean/Median Imputation: If the distribution is relatively normal, use the mean; if skewed, use the median. This is straightforward but may not capture the variability.
  - Predictive Modeling: Use regression models to predict missing values based on other features (e.g., Time\_of\_Transaction, Device\_Used).
  - K-Nearest Neighbors (KNN): Impute based on the average of the nearest neighbors in the feature space.
- 2. Time\_of\_Transaction (2552 missing values)
  - Mean/Median Imputation: Similar to Transaction\_Amount, use mean or median based on the distribution.
  - Time-based Imputation: If the data has a temporal component (e.g., hourly, daily), consider using the average transaction time for that specific hour/day.
  - Predictive Modeling: Use other features to predict the time of transaction.
- 3. Device\_Used (2473 missing values)
  - Mode Imputation: Since this is a categorical variable, using the mode (most frequent category) is a common approach.
  - Predictive Modeling: Use other features to predict the device used.
  - Create a New Category: If the missing values are significant, consider creating a new category (e.g., "Unknown") to indicate missing data.

Users can see missing values insights after dataset health check.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Baseline Model Check' section of the AI Workbench. On the left, a sidebar includes 'Task Notification Hub' (No active background tasks), 'Admin Session Active' (Debug: Role:admin, Perms:9), and a 'Navigation' menu with options like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow (selected), LLM Fine Tuning, API Hub, and Admin Console.

The main area displays the following data:

- 6. Data Augmentation:** A note stating: "If applicable, you can also explore data augmentation techniques to create variations of the existing fraudulent instances, which can help in increasing the diversity of the minority class. By applying one or a combination of these techniques, you can improve the performance of your model on the imbalanced dataset."
- 1.4 Baseline Model Check**: Includes a 'TRAIN BASELINE MODEL' button, 'Preview Accuracy' (95.1%), 'Preview F1-Score' (0.93), and a 'Confusion Matrix (All)' table:

		Predicted Label	
Actual Label	Normal		Fraud
	Fraud	Normal	743
Fraud	6	743	750
Normal	0	743	743
- Top Drivers (Feature Importance)**: A horizontal bar chart showing feature importance values for various features like Transaction\_Amt, Account\_Age, Total\_Trans, Number\_of\_Trans, and Previous\_Fraud. The x-axis ranges from 0.00 to 0.45.
- 1.5 Generate Final Dataset**: Includes a 'PROCESS & SPLIT DATA' button.

Users can do baseline model check to ensure current dataset features importance and then can execute to split and generate a dataset for training.

The screenshot shows the 'End-to-End ML Workflow' interface. The 'Step 2: Auto-Train Fraud Models' section is highlighted. It includes a note: "Train baseline models (Logistic Regression & Random Forest) using the dataset created in Step 1." Below this, there are sections for 'Server Save Directory' (data/models), 'Enable Hyperparameter Optimization (Slower but accurate)', and 'Advanced Model Configuration' for Random Forest, Gradient Boosting, and Logistic Regression. Each configuration includes sliders for parameters like 'RF Trees (n\_estimators)', 'RF Max Depth', 'Learning Rate', and 'Regularization Strength (C)'. At the bottom is a 'TRAIN MODELS NOW' button.

Before train the mode, users can manually adjust the model parameters and then click "Train Models now".

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Metrics' section of the AI Workbench. At the top, a green banner displays 'Training complete! Best model: random\_forest'. Below this, the 'Metrics' heading is followed by three sections: 'random\_forest', 'gradient\_boosting', and 'logistic\_regression'. Each section contains a summary table with accuracy, precision, recall, F1-score, and support values, along with detailed classification reports for both Class 0 and Class 1.

Model	accuracy	precision	recall	F1-score	support
random_forest	0.9426229598196722	0.94	1.00	0.97	1158
gradient_boosting	0.9499836866557377	0.47	0.50	0.48	1228
logistic_regression	0.9426229598196722	0.89	0.94	0.91	1228

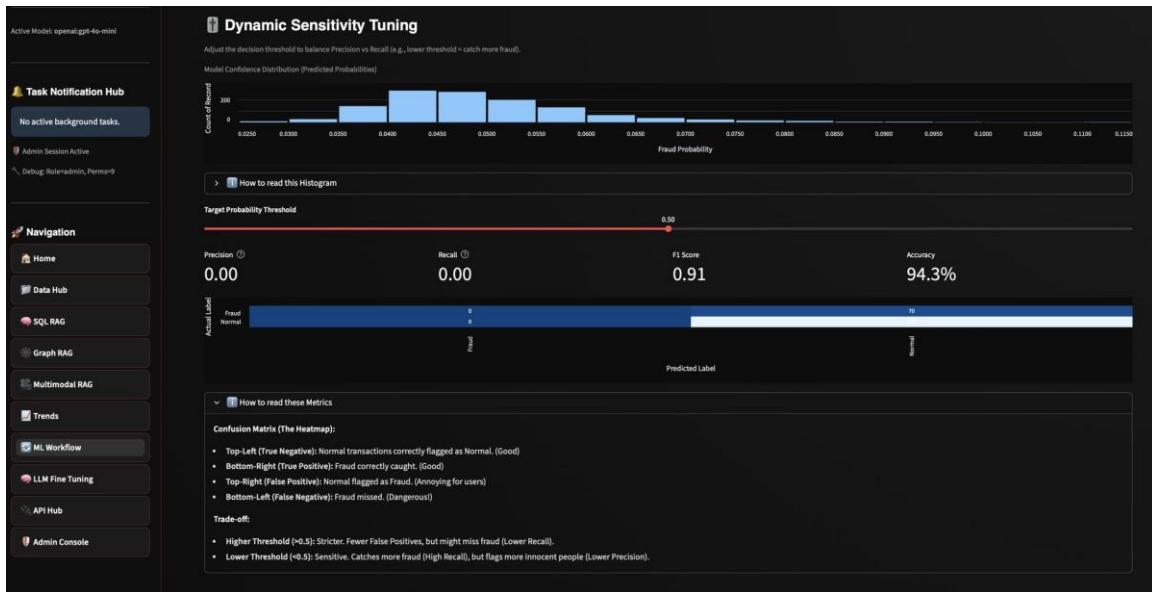
After training the mode, users can see the model metrics.

The screenshot shows the 'ANALYZE PERFORMANCE' page. It compares the Random Forest (RF) and Logistic Regression (LR) models. Both models have an accuracy of approximately 0.9426. The RF model has a higher ROC AUC (0.4895) compared to LR (0.4987). The page also includes a 'Comparison' section with three numbered points explaining the performance differences.

- Accuracy: Both models have the same accuracy of approximately 0.9426, indicating that they perform similarly in terms of overall correct predictions.
- Precision, Recall, and F1-score:
  - Both models have a precision, recall, and F1-score of 0.0 for the positive class (Class 1). This indicates that neither model is able to correctly identify any instances of fraud (Class 1).
  - For the negative class (Class 0), both models perform well, achieving high precision (0.94), recall (1.00), and F1-score (0.97).
- ROC AUC:

There is overview of Metrices shown under neath.

## Introduction to Fraud detection AI Workbench (version pro)



Users can choose dynamically sensitive tuning.

## 11. UI8 - LLM Fine Tuning

### 11.1 Functionality Design

Designed to bring localized, secure Large Language Model optimization to the analyst without requiring cloud execution (preventing PII/data leakage). It utilizes Low-Rank Adaptation (LoRA) via Apple's MLX (or PyTorch) framework to modify model adapters in a fraction of traditional compilation time. The functional architecture isolates the dataset generation (Extracting historical investigator prompts or using teacher-LLM synthetics) from the multi-threaded inference testbed, ensuring empirical validation of model hallucination reduction.

### 11.2 UI Design

A highly structured 'Command Center' consisting of four distinct tabs: Collect, Review, Train, and Test. The UI balances technical command configurations (LoRA rank, Alpha, Batch limits) with intuitive data viewing capabilities. The 'Test' tab features a dual-chat split-pane, rendering identical subsequent prompts to both a 'Base Model' and a 'Fine-Tuned Model' concurrently for direct A/B visual comparison.

### 11.2 User Manual

Step 1: Use the 'Collect Dataset' tab to securely aggregate Chat/RAG logs, or synthetically generate instructional JSONL datasets using an LLM teacher.

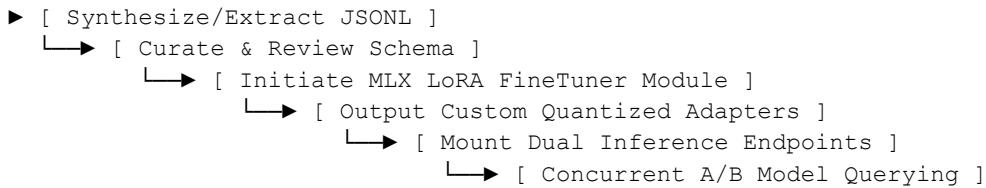
Step 2: In 'Review', curate the rows to ensure high-quality question/answer pairs and delete anomalous examples.

Step 3: In the 'Train' tab, select a Base Model (e.g., LLaMA), configure LoRA parameters, and execute the localized MLX training cycle.

## Introduction to Fraud detection AI Workbench (version pro)

Step 4: Launch the 'Dual-Chat Test' to interrogate both models simultaneously, verifying that the new heavily-weighted fraud knowledge adheres properly.

### 11.3 Logic Workflow



### 11.4 Demo

In LLM Tuning lab, there are 4 steps to do:

- Data collection
- Dataset review
- Fine tuning
- Model testing

In the Data collection, user or data scientist needs to customize data entry one by one to evaluate risk degree.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the AI Workbench interface with the following sections:

- Task Notification Hub**: Shows "No active background tasks.", "Admin Session Active", "Debug: Roleadmin, Perm:rw", and a search bar for Transaction ID.
- Manual Selection**: A card for transaction T15 showing a probability of 4.6% and a fraud level of LOW RISK.
- AI Quick Recommendation**: A card for "Not Fraud" with a green "High" confidence level and a 25% fraud vote.
- Transaction Details**: A table of transaction data including Transaction\_ID, User\_ID, Transaction\_Amount, Transaction\_Type, Time\_of\_Transaction, Device\_Used, Location, and various fraud-related metrics.
- ML Confidence Metrics**: A section showing ML confidence metrics.

Based on selected data collection, AI gives suggestions for risk recommendations.

The screenshot shows the AI Workbench interface with the following sections:

- Task Notification Hub**: Shows "No active background tasks.", "Admin Session Active", "Debug: Roleadmin, Perm:rw", and a search bar for Transaction ID.
- Ensemble Voting Breakdown**: A table showing Fraud votes (1/4), Safe votes (3/4), and Uncertain (0/4). Model Probability is 4.6%.
- Your Expert Feedback**: A form for providing expert analysis, explaining why it's or isn't fraud, and what action to take.
- Fraud Category**: A dropdown menu for "Fraud Category (AI suggests: Not Fraud)" with options: Not Fraud, Account Takeover, Identity Theft, Card Not Present, Synthetic Identity, and Other. The "Not Fraud" option is selected. An "AI Quality" dropdown is set to "Good". A progress bar shows "Examples 107/100". Buttons for "SAVE CURRENT" and "CLEAN DATASET" are present.
- Feedback Message**: A message at the bottom right: "You've reached 100 examples! Ready to fine-tune in Tab 3!"

Based on knowledge, data scientist manually evaluates the risk scale and give his/her decision. Data scientist can scan data entry one by one from the data collection and give decision for fraud yes or not.

## Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the 'Dataset Review' section of the LLM Fine-Tuning Lab. On the left, a navigation sidebar lists various tools: Task Notification Hub, Admin Session Active, Navigation (Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, Admin Console), and Help & Documents. The main area has a header 'LLM Fine-Tuning Lab' with a sub-header 'Fine-tune language models to improve fraud analysis reasoning using your expert feedback.' Below this is a progress bar showing steps 1. Data Collection, 2. Dataset Review (which is active), 3. Fine-Tuning, and 4. Model Testing. The 'Dataset Review' section contains a table with columns 'Select Dataset' (containing '107 samples (107 samples)'), 'RENAME', 'CLONE', and 'DELETE'. A message 'Loaded 107 examples from "107 samples"' is displayed. The 'Statistics' section shows 'Total' 107, 'Categories' 4, and 'Avg Length' 620. It includes two horizontal bar charts: 'Fraud Categories' (Net Fraud, Synthetic Identity) and 'Quality Ratings' (Excellent, Good, Fair, Poor, Bad).

In data review, data scientist will review samples for LLM fine tuning.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot displays two main sections of the AI Workbench interface:

- LLM Fine-Tuning Lab**: This section is titled "Fine-Tuning" and "Training Configuration". It shows a "Training Dataset" table with one entry: "New Default Dataset (221 samples)" with 221 samples. Below this is a "Model Registry" table listing four trained models:

Model Name	Dataset Source	Base LLM	Epochs	Created
Default Model	New Default Dataset	mlx-community/Llama-3.2-1B-Instruct-4bit	3	2025-12-19 06:03
Ollama_llama3_8b_expert	100 samples	ollama:llama3:8b	N/A	2026-01-02 15:59
Ollama_qwen3_8b_expert	100 samples	ollama:qwen3:8b	N/A	2026-01-02 16:00
Ollama-qwen3_8b	107 samples	ollama:qwen3:8b	N/A	2026-01-02 16:03
- MLX Configuration**: This section includes "Select Training Approach" (Option 1: MLX Fine-Tuning (True Model Training), Option 3: Ollama Expert Prompts (Smart Prompting)), "MLX Configuration" (Base Model: mlx-community/Llama-3.2-1B-Instruct-4bit, Epochs: 3, Custom Version Name: e.g., Llama-3.2-Fraud-v1), "Training" (Training Iterations: 300, EXPORT DATASET FOR MLX button), and a terminal command input field containing: `mlx_lm.lora --model mlx-community/Llama-3.2-1B-Instruct-4bit --train --data data/lm/mlx_dataset --iters 300`. A large red "START TRAINING" button is at the bottom.

User can start training LLM model. The LLM model can select local installed LLM. In the Admin-Console, administrator can install local LLM into Ollama.

# Introduction to Fraud detection AI Workbench (version pro)

The screenshot displays the Fraud detection AI Workbench interface, specifically the LLM Fine-Tuning Lab section. The left sidebar contains a navigation menu with various options like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning (which is currently selected), API Hub, Admin Console, and Help & Documents.

The main content area is titled "LLM Fine-Tuning Lab" and includes a sub-section titled "Model Testing". It features a "Comparison Settings" panel where users can choose between "Inference Mode" (selected), "Simulated (Instant)", or "Real (Slower, Actual Models)". Below this is a "Select Test Transaction" section showing a transaction with ID #1: Prob: 0.01% | Amt: 208.81 | Age: 16.

On the right, there's a "MLX Version" dropdown set to "Default Model (2025-12-19 06:03)" and a summary card for the "Base vs MLX Fine-Tuned" comparison:

MLX Metadata	Dataset: New Default Dataset	Trained: 2025-12-19 06:03
	Base: mix-community/Llama-3.2-1B-Instruct-4bit	Epochs: 3

Below this, the "Measured Differences" section compares three models:

Model	Overall Score	Notes
Base Model	3.0/10	Lacks specificity
MLX Fine-Tuned	8.5/10	Specialized & detailed
Ollama Expert	9.0/10	Best balance

Each model row includes a "Risk Factors Identified" section with counts for "vs Ollama" and "vs MLX". The "Ollama Expert" row also includes a "What Expert Models ADD:" section with counts for "MLX Data Points" and "Ollama Red Flags".

A note at the bottom states: "High-Risk Transaction → Use MLX or Ollama".

At the very bottom, a note says: "Note: Metrics shown are simulated. For real inference, complete training in Tab 3."

After LLM has been fined tuning, it becomes a new model in the local installed LLM collection. Users can compare LLM in the different fine tuning mode. There are 3 compare modes:

- Base or original model
- MLX Fine-Tuned, which is cloud based LLM training
- Ollama Expert, which is a local LLM training

## Introduction to Fraud detection AI Workbench (version pro)

12. UI9 - API Interaction Hub

## 12.1 UI Design

A highly formalized developer portal aesthetic featuring terminal-dark code block structures with rapid 'Copy' clipboard logic. It seamlessly auto-embeds the native FastAPI Swagger documentation sandbox into an HTML frame.

## 12.2 User Manual

Step 1: Administer the generation of a long-lived API execution token.

Step 2: Review the dynamically updating parameter requirements.

Step 3: Utilize the generated cURL/Python integration scripts to tie external banking infrastructure directly into Sentinel's scoring pipeline, pushing live transactions for millisecond ML evaluation.

## 12.3 Logic Workflow



## 12.4 Demo

The screenshot displays the API Interaction Hub interface, which is a live testing console for all Sentinel Pro FastAPI backend endpoints. The left sidebar includes sections for SENTINEL (Fraud Detection AI Workbench), User Session (Logged in as: admin, Role: admin, Logout), Global Configuration (Active Model: openai:gpt-4-mini, API Keys), and Task Notification Hub (No active background tasks). The main content area shows several API endpoints:

- /auth/token**: A POST endpoint for logging in and getting a JWT bearer token. It includes fields for Username (yahuohu68) and Password, a red LOGIN button, and a green success message with a duration of 0.39s.
- /keys/**: A GET endpoint for listing all active API keys. It shows a table with columns for Key ID, Key Name, and Revoked status.
- /keys/generate**: A POST endpoint for generating a new named API key. It has a Key Name field set to "My Service Key" and a GENERATE KEY button.
- /keys/id**: A DELETE endpoint for revoking an API key by ID. It has a Key ID to Revoke field set to 1 and a REVOKE KEY button.
- /auth/register**: A POST endpoint for registering a new user account. It includes fields for Username and Email (optional), with a sample value of user@example.com.

In API Interaction Hub, there are 6 tasks for user choosing:

- Identity & Auth
  - Ingestion
  - Intelligence
  - ML Models

## Introduction to Fraud detection AI Workbench (version pro)

- Admin & Infrastructure
- Health

In Identity & Auth, user can test API connection for Identity or Auth. For example, for login, user simply input username and password and click “Login”, user can see cURL command and JSON response.

The screenshot shows the API Interaction Hub interface. On the left, there's a sidebar with sections like User Session, Global Configuration (API Keys), Task Notification Hub (No active background tasks), and Admin Console. The main area is titled "API Interaction Hub" and has tabs for Identity & Auth, Ingestion, Intelligence, ML Models, Admin & Infra, and Health. The Ingestion tab is selected. It shows a table of supported file types and their behaviors:

Extension	SQL Table	Graph Store	Knowledge Base
CSV	<input checked="" type="checkbox"/> auto-create table	<input checked="" type="checkbox"/> doc node	<input checked="" type="checkbox"/> rebuild
PDF	—	<input checked="" type="checkbox"/> text extracted	<input checked="" type="checkbox"/> rebuild
PNG/JPG	—	<input checked="" type="checkbox"/> image node	<input checked="" type="checkbox"/> rebuild
MP3/WAV	—	<input checked="" type="checkbox"/> transcript	<input checked="" type="checkbox"/> rebuild
MP4	—	<input checked="" type="checkbox"/> frames + transcript	<input checked="" type="checkbox"/> rebuild

Below the table, there's a "Select file" button, a "Drag and drop file here" field with a 350MB limit, a "Browse files" button, and an "INGEST FILE" button. To the right, there's a "How it works" section with a list of steps:

- Parses `CREATE TABLE` statements and drops existing tables before re-creating
- Executes the full script via `conn.executescript()`
- Stores the SQL as a document in the Graph Store
- Syncs schema changes to the graph

At the bottom right is an "EXECUTE SQL SCRIPT" button.

In Ingestion, users can test uploading data files or executing SQL files.

The screenshot shows the API Interaction Hub interface. The sidebar includes sections like Navigation (Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, Admin Console), Help & Documents, and API Configuration (Base URL: http://fastapi:8000). The main area is titled "API Interaction Hub" and has tabs for Multi-Agent Pipeline and Legacy SQL/NLQ. The Legacy SQL/NLQ tab is selected. It shows a "Request Schema" section with a JSON code snippet:

```
    {  
        "question": "string \u2013 natural language question",  
        "llm_id": "string \u2013 e.g., openai:gpt-4o-min",  
        "k_candidates": "int (1-5) \u2013 retrieval candidates",  
        "bypass_agents": "bool \u2013 skip deep synthesis (faster, avoids timeouts)",  
        "rebuild_kb": "bool \u2013 force KB rebuild before query"  
    }
```

Below this is a "Question" section with the text "Which merchants have suspicious high-volume transactions?". Underneath is an "LLM" dropdown set to "openai:gpt-4o-min", a "Speed Mode (bypass agents)" checkbox, a "Rebuild KB" checkbox, and a "Candidates (k)" slider set to 3. At the bottom is a "RUN AGENT PIPELINE" button.

In Intelligence, users can test API connection for LLM. User simply input question to check JSON response.

## Introduction to Fraud detection AI Workbench (version pro)

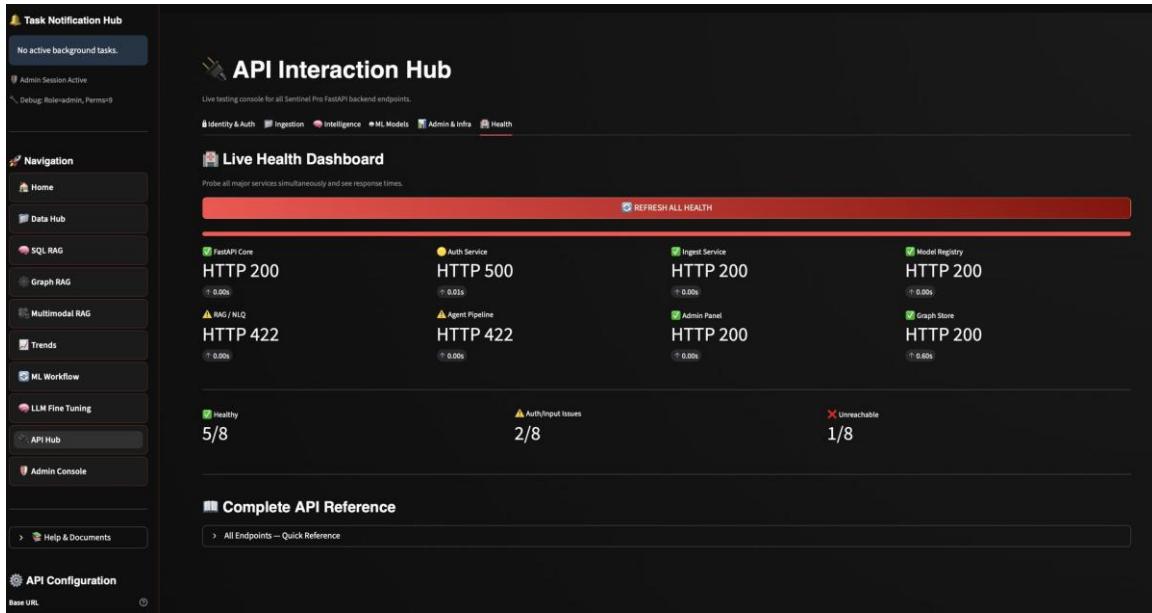
The screenshot shows the API Interaction Hub interface. On the left, there's a navigation sidebar with various options like Home, Data Hub, SQL RAG, Graph RAG, Multimodal RAG, Trends, ML Workflow, LLM Fine Tuning, API Hub, and Admin Console. The main area is titled "API Interaction Hub" and has tabs for Identity & Auth, Ingestion, Intelligence, ML Models, Admin & Infra, and Health. Under the "ML Models" tab, there are sections for Model Registry, ML Scoring, and Report Generation. A "LIST ML MODELS" button is shown with a cURL command below it. To the right, there's a "DISCOVER LLMs" section with a "Discover available LLMs (Ollama + cloud)" button. The central part of the screen displays a JSON response from the "/models/list" endpoint, listing two models: "default\_model" and "ollama\_llama3\_8b\_expert".

In ML Models, users can test API connections for LLM, such as list ML models, or discover LLMs.

The screenshot shows the API Interaction Hub interface with the "Admin & Infra" tab selected in the top navigation bar. The left sidebar remains the same. The main area is titled "API Interaction Hub" and has tabs for User Management, Role & Permissions, Storage & Graph, and Admin & Infra. Under the "User Management" tab, there are sections for User Management, Role & Permissions, Storage & Graph, and Admin & Infra. A "LIST ALL USERS" button is shown with a cURL command below it. To the right, there are four separate panels for updating user roles, emails, and usernames. Each panel has a "User ID" input field set to "1" and a "New [Role/Email/Username]" input field. Buttons for "UPDATE ROLE", "UPDATE EMAIL", and "UPDATE USERNAME" are visible in each panel respectively.

In Admin & Infra, user can test API connections for admin console and infrastructure management.

## Introduction to Fraud detection AI Workbench (version pro)



In the Health page, user can do live health check for API connections.

## 13. UI10 - Admin Console

### 13.1 UI Design

An operational control matrix exclusively protected by strict JWT RBAC scoping. Employs highly organized `st.data\_editor` elements for bulk configuration mapping. All changes exhibit immediate hot-reloading behaviors across universal active sessions.

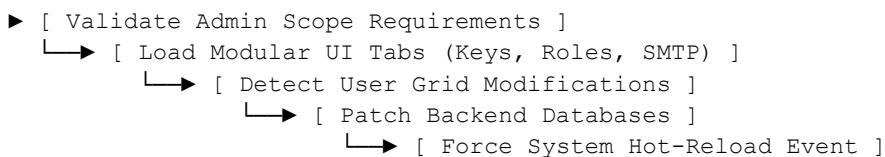
### 13.2 User Manual

Step 1: Only accessible by authenticated administrators.

Step 2: Provide specific OpenAI, DeepSeek, or Cloud API tokens; these map dynamically and overwrite `.env` states instantly.

Step 3: Provision new analyst user accounts, enforce mandatory password resets, and adjust viewing permissions per UI component via toggles.

### 13.3 Logic Workflow



### 13.4 Demo

In Admin Console, there are 7 tabs to manage below system settings:

- API Keys
- User Management

## Introduction to Fraud detection AI Workbench (version pro)

- Role Management
- External Credentials
- Local LLMs
- System Stats
- SMTP Settings

The screenshot shows the Admin Console interface. On the left, there's a sidebar with sections like User Session, Global Configuration (API Keys selected), Task Notification Hub, and Navigation. The main content area is titled "Admin Console" and "Centralized management for Fraud Lab security and system settings". It has tabs for API Keys, User Management, Role Management, External Credentials, Local LLMs, System Stats, and SMTP Settings. The "API Keys" tab is active, showing a section for "Technical API Keys". It includes a button to "Generate New API Key" and a table titled "Active Keys" listing several keys with their creation dates and delete icons.

In API keys page, admin can generate new API key and delete existing API key(tokens).

ann

The screenshot shows the Admin Console interface. The sidebar is identical to the previous one. The main content area is titled "User Management" and "Directly manage application users and their access roles". It includes a search bar, a filter for username or role, and a table listing users with columns for ID, Username, Email, Role, Security, and Delete. The table shows six users: admin, guest, scientist1, stephane@qubec, david@ontario, and david@demo.com, each with their respective details and edit/delete buttons.

In User Management page, admin can change username, password, role and email address. Also, admin can delete user.

## Introduction to Fraud detection AI Workbench (version pro)

The screenshot shows the Admin Console interface. On the left is a sidebar with sections for User Session (Logged in as: admin, Role: admin), Global Configuration (API Keys, Select Active LLM: openai/gpt-4o-mini, Active Model: openai/gpt-4o-mini), and Task Notification Hub (No active background tasks). The main content area is titled "Role Management" and contains a sub-section for "Role: guest". It lists "Permissions (Allowed Pages)" with checkboxes for various pages like Data\_Hub, SQL\_RAG\_Assistant, Graph\_RAG\_Assistant, Multimodal\_RAG\_Assistant, Trends\_and\_Insights, ML\_Workflow, LLM\_Fine\_Tuning, API\_Interaction, and Admin\_Console. A "SAVE PERMISSIONS" button is at the bottom right. Below this is another section for "Role: data\_scientist" and "Role: admin".

In Role Management page, admin set which page the role can access. System has 3 default roles:

- Guest
- Data\_Scientist
- Admin

The screenshot shows the Admin Console interface. The sidebar is identical to the previous one. The main content area is titled "External Credentials" and contains two sections: "Kaggle API" and "GitHub API". Under "Kaggle API", there are fields for "Kaggle Username" and "Kaggle Key", with "SAVE KAGGLE CREDENTIALS" and "TEST KAGGLE CONNECTION" buttons. Under "GitHub API", there is a field for "GitHub Token (Optional, for private repos/higher limits)" with "SAVE GITHUB TOKEN" and "TEST GITHUB CONNECTION" buttons.

In External Credentials page, admin can set credentials for Kaggle and GitHub API access token. After completing setting, admin can save and test connection.

## Introduction to Fraud detection AI Workbench (version pro)

Model Name	Size	Modified	Action
nomic-embed-text:latest	0.26 GB	2026-02-01	
llama3:8b	4.34 GB	2025-12-17	
llama3.2:latest	1.88 GB	2025-12-03	
open3:8b	4.87 GB	2025-12-03	

In Local LLMs page, admin can review and delete local installed LLMs.

Model Name	Size	Modified	Action
nomic-embed-text:latest	0.26 GB	2026-02-01	
llama3:8b	4.34 GB	2025-12-17	
llama3.2:latest	1.88 GB	2025-12-03	

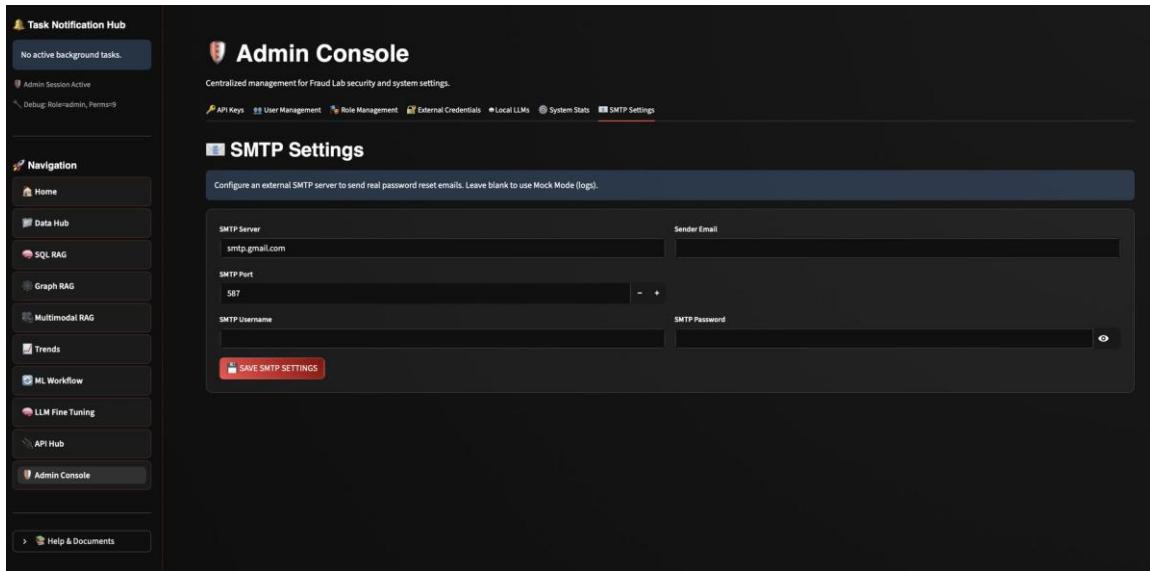
Available Models

- Llama 3 (4.7 GB)
- Mistral (4.1 GB)
- Gemma 2B (1.5 GB)
- Phi-3 Mini (2.3 GB)
- Qwen 0.5B (550 MB)
- TinyLlama (600 MB)
- Llama 3 (4.7 GB)

**PULL MODEL**

Admin can install LLM with small billion parameters at the local Ollama. System default defines a list of LLMs for download and installation.

## Introduction to Fraud detection AI Workbench (version pro)



Admin can setup SMTP for sending email if user get notification for reset or forget password.

## 14. Show Demo

### 14.1 Description

A dedicated function located in the sidebar designed to present an interactive, pre-populated demonstration of the platform's capabilities without requiring user data upload.

### 14.2 Steps

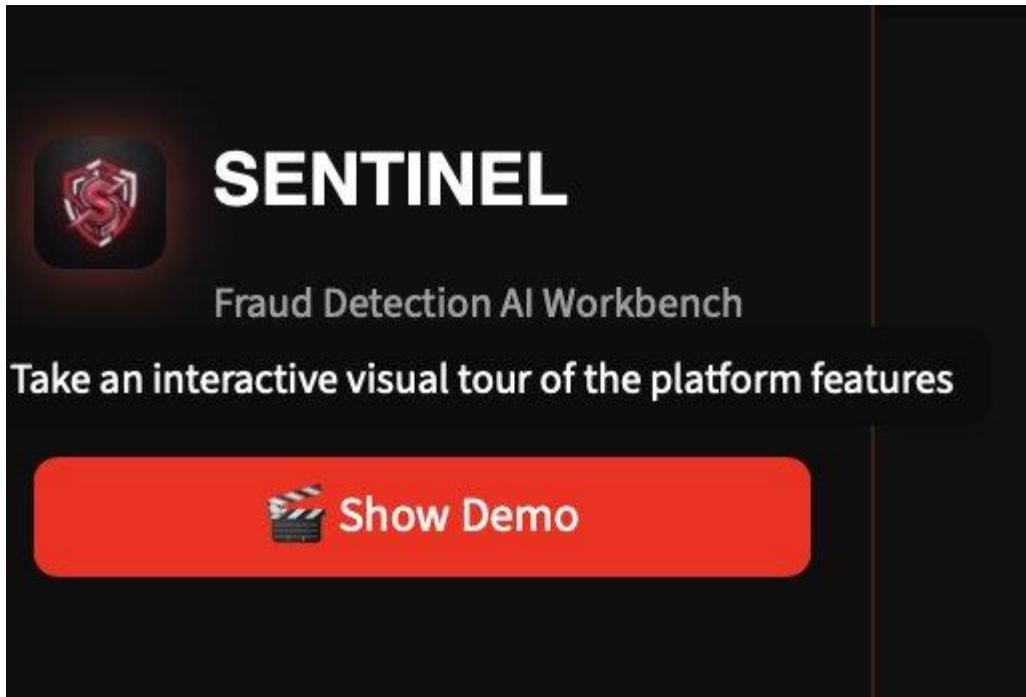
- Step 1: Locate the 'Show Demo' toggle or button in the main application sidebar.
- Step 2: Activate the demo mode to generate and load synthetic demonstration data into the application.
- Step 3: Interact with the populated charts, tables, and graphs to explore the platform's features safely.

### 14.3 Impacts to the System

When activated, the Show Demo function bypasses live data fetching and database queries, injecting static synthetic payloads directly into the Session State. The system operates entirely in-memory for the demo session, temporarily suspending external API calls and persistent writes, ensuring that no experimental changes corrupt the underlying Neo4j or SQLite production data stores.

### 14.4 Demo

## Introduction to Fraud detection AI Workbench (version pro)



In the sidebar, there is a button to provide an interactive visual tour of the platform features.

### 15. Load Demo Data

#### 15.1 Description

A privileged administrative action designed to bootstrap the system by seeding the underlying databases with a comprehensive suite of interconnected, synthetic fraud data. This allows for immediate, full-scale testing of the platform's analytical capabilities without requiring manual data ingestion.

#### 15.2 Steps and Location

- Step 1: Authenticate as a user with 'Administrator' privileges.
- Step 2: Navigate to the 'Admin Console' via the main sidebar.
- Step 3: Locate the 'Load Demo Data' button within the Database Management or Provisioning section of the console.
- Step 4: Click the button and confirm the action to initiate the initialization pipeline.

#### 15.3 Impacts to the System

When executed, this function triggers a multi-stage background pipeline that profoundly alters the system state:

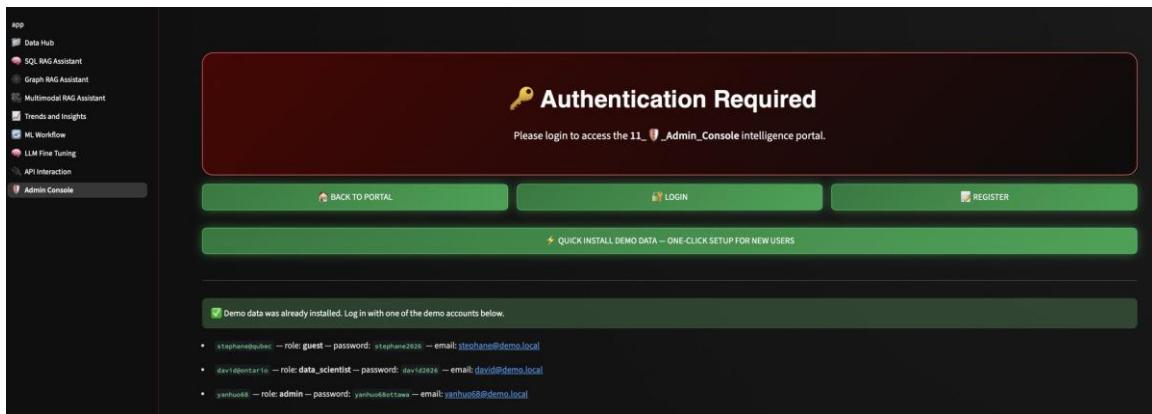
- Relational Database (SQLite/PostgreSQL): Injects thousands of synthetic transaction records, user profiles, and account details representing typical and fraudulent activity patterns.

## Introduction to Fraud detection AI Workbench (version pro)

- Graph Database (Neo4j): Constructs a complex web of interconnected nodes (e.g., Users, IP Addresses, Devices, Bank Accounts) and semantic edges (e.g., 'TRANSFERRED\_TO', 'LOGGED\_IN\_FROM'), explicitly modeling known fraud syndicates for Graph RAG analysis.
- Vector Store (FAISS): Embeds pre-generated textual scenarios, simulated case notes, and policy documents, populating the semantic index to enable context-aware similarity searches and Document RAG functionalities.

Warning: This action writes directly to persistent storage and may overwrite or conflict with existing sandbox data.

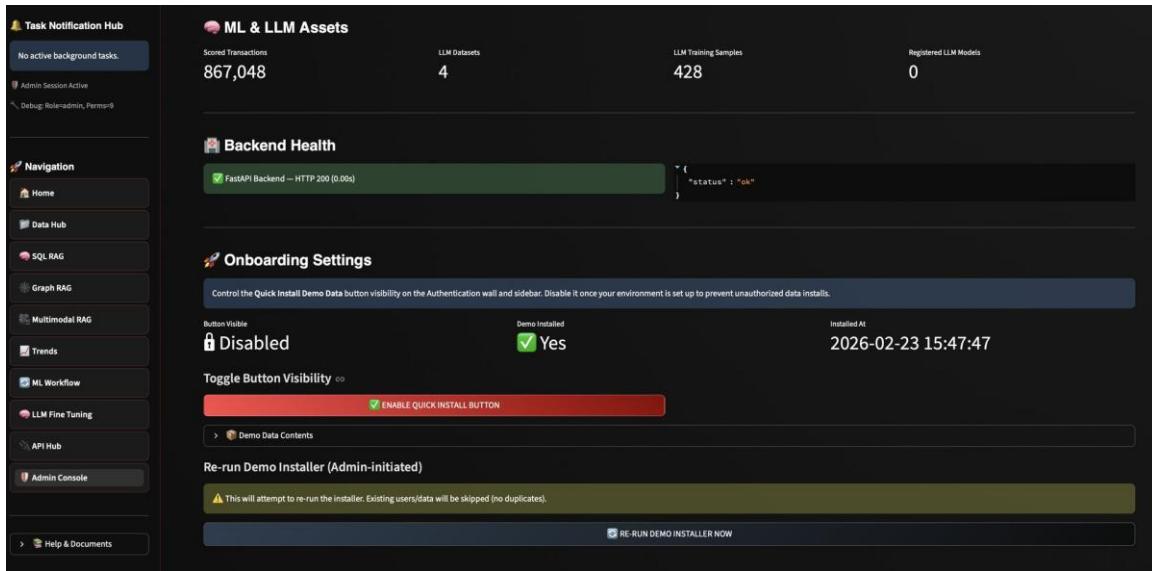
### 15.4 Demo



When user initial comes to system, system provides a convenient way to load pre-prepare dataset, including below:

- Create 3 users
  - Each user has a different role. The role includes guest(least privilege), data\_scientist and admin.
- Upload fraud-detect-dataset.csv
- Execute 4 relational tables into the SQLite

## Introduction to Fraud detection AI Workbench (version pro)



In the Admin Console, admin can reset Load Demo button enabled after user has already clicked this button.

## 16. Dependency References

### 16.1 FastAPI

Utilized as the core asynchronous backend API router. Chosen for its automatic OpenAPI integration, high throughput, and strict Pydantic data validation model.

### 16.2 Streamlit

The primary frontend reactive visualization framework. Chosen to accelerate ML tool delivery by eliminating complex React/Vue boilerplate while maintaining interactive graphs.

### 16.3 SQLAlchemy

An Object Relational Mapper (ORM). Facilitates database-agnostic modeling, ensuring Sentinel can map securely across simple flat SQLite files or enterprise PostgreSQL clusters.

### 16.4 Neo4j (urllib3/neo4j-driver)

A dedicated graph database. Integral for calculating multi-hop relationships (fraud rings) that deeply struggle in standard row-based SQL structures.

### 16.5 FAISS

Facebook AI Similarity Search. A highly optimized C++ vector library wrapped in Python enabling instantaneous semantic similarity retrievals across billions of embedded document chunks.

## Introduction to Fraud detection AI Workbench (version pro)

### 16.6 SQLGlot

An incredibly efficient SQL parser. Necessary for parsing dangerous LLM-generated strings into AST representations to programmatically guarantee the absence of destructive DROP/ALTER statements before engine execution.

### 16.7 PyVis / NetworkX

Python-native graph visualization suites. Translates raw underlying neo4j edge coordinates into human-readable, physics-simulated node clusters seamlessly overlaid in Streamlit.

### 16.8 BCrypt

A computationally heavy cryptographic hashing algorithm explicitly chosen to counteract brute-force hardware attacks when storing registered investigator passwords.

### 16.9 Ollama

An underlying local inference engine wrapper. Enables Sentinel to rapidly pull, mount, and execute quantized open-weight models locally safely entirely off-network.

## 17. Glossary

### RAG (Retrieval-Augmented Generation):

A methodology where an LLM is paired with an external knowledge retrieval mechanism (like Vector DBs) to ground its responses in factual, enterprise-specific context, eliminating hallucination.

### Knowledge Graph:

A structured taxonomy of concepts and relations (Nodes and Edges). Essential for depicting complex fraud rings where individuals, IP addresses, and bank accounts overlap.

### LLM (Large Language Model):

A massive neural network trained on vast quantities of text. Capable of translation, summarization, logical bridging, and code compilation (e.g., text-to-SQL logic).

### Fine-Tuning (LoRA):

Low-Rank Adaptation. A computationally efficient mathematical shortcut to retrain a massive LLM on specific behavior without altering its fundamental parameter structure.

### RBAC (Role-Based Access Control):

A security paradigm where access logic is denied inherently unless explicitly permitted via a matrix mapped to the authenticated user's registered job function.

### JWT (JSON Web Token):

## **Introduction to Fraud detection AI Workbench (version pro)**

A cryptographically signed, compact URL-safe execution credential passed between the client and server to verify session integrity asynchronously.

### **Vector Embeddings:**

Array structures representing the profound semantic meaning of text/images as mathematical distances, allowing search algorithms to identify context similarities regardless of explicit keyword matches.

### **SHAP (SHapley Additive exPlanations):**

A game-theoretic ML model evaluation technique that quantifies and visually explains the exact logic path an algorithmic decision tree utilized to calculate a fraud-score output.