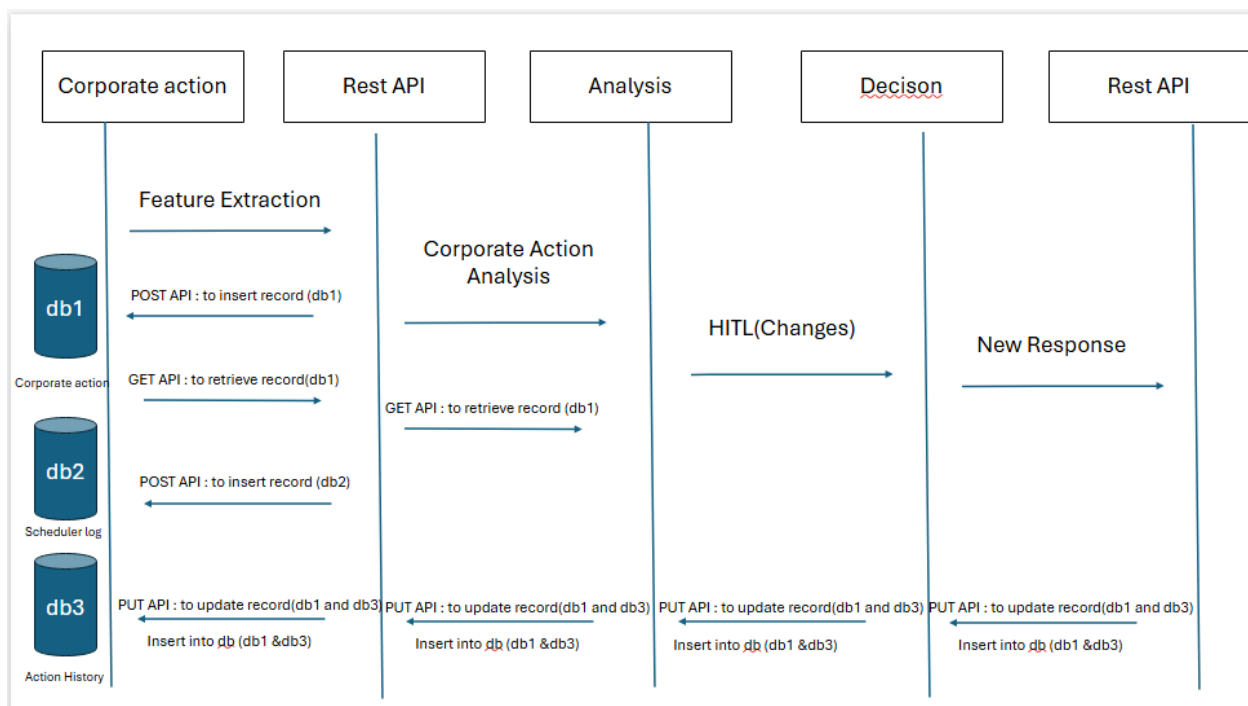


Corporate Action Data Flow and Database Schema

Table Names

1. Corporate_Actions
2. Scheduler_Logs
3. Action_History



Data Flow Process

1. **Receiving Corporate Action:**
 - a. When a new corporate action is received, the system extracts feature and sends them to the backend via a REST API to update the **Corporate_Actions** table in the database (db1).

2. **Sending Metadata:**
 - a. The system also sends metadata information related to the corporate action via a REST API to update the **Scheduler_Logs** table.
3. **Retrieving Records:**
 - a. The system retrieves the corresponding records via a REST API (GET API) for analysis and display on the UI.
4. **Context-Aware Analysis:**
 - a. A context-aware analysis determines whether the corporate action should be verified or not by making necessary changes.
5. **Human in the Loop (HITL):**
 - a. To verify the analysis, a decision-making process involving human intervention (HITL) is introduced.
6. **Updating Records:**
 - a. If changes are made, the system updates the **Corporate_Action** table and **Action_History** table via a REST API (PUT API).

Database Schema

PostgreSQL Database Credentials

- **Hostname:** localhost
- **Port:** 5432
- **User:** your_username
- **Password:** your_password
- **Database Name:** yourdbname

Table Names

4. Corporate_Actions
5. Scheduler_Logs
6. Action_History

Table Schemas

Corporate_Actions Table

```
CREATE TABLE Corporate_Actions (  
    id SERIAL PRIMARY KEY,
```

```

Company VARCHAR(255),
Corporate_Action VARCHAR(255),
Date_Announcement TIMESTAMP,
Source VARCHAR(255),
Extracted_Information TEXT,
Status VARCHAR(50) CHECK (Status IN ('Verified', 'Not Verified')),
Insertion_Date_Time TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
Modified_Date_Time TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

```

Scheduler_Logs Table

```

CREATE TABLE Scheduler_Logs (
    id SERIAL PRIMARY KEY,
    Scheduler_Name VARCHAR(255),
    Source VARCHAR(255),
    Date_Time_Scheduler_Ran TIMESTAMP,
    Total_Records_Fetched INT,
    Time_Taken INT,
    Status VARCHAR(50) CHECK (Status IN ('Success', 'Failure'))
);

```

Action_History Table

```

CREATE TABLE Action_History (
    id INT,
    Extracted_Information_Before TEXT,
    Extracted_Information_After TEXT,
    Modified_Date_Time TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

```

API Endpoints

Corporate_Actions Table

- **POST:** <http://127.0.0.1:8000/insert-corporate-action/>
 - To enter data into the Corporate_Actions table.
 - Example Request Body:

```
{
  "Company": "Tech Innovators Inc.",
  "Corporate_Action": "Merger Announcement",
  "Date_Announcement": "2025-01-15T10:30:00Z",
  "Source": "Official Press Release",
  "Extracted_Information": "Tech Innovators Inc. announced a merger with Future Solutions Ltd. to expand their market reach and product offerings.",
  "Insertion_Date_Time": "2025-01-15T10:35:00Z",
  "Modified_Date_Time": "2025-01-15T10:35:00Z"
}
```

- **PUT:** <http://127.0.0.1:8000/update-corporate-action/>
 - To update records in the Corporate_Actions table based on the primary key.
 - Example Request Body:

```
{
  "id": 4,
  "Extracted_Information": "this is updated"
}
```

- **GET:** <http://127.0.0.1:8000/get-records-today>
 - To select records where InsertionDateTime is today.
- **GET:** <http://127.0.0.1:8000/get-records-before-today/>
 - To select records where InsertionDateTime is before today.

Scheduler_Logs Table

- **POST:** <http://127.0.0.1:8000/insert-scheduler-log/>
 - To insert new values into the Scheduler_Logs table.
 - Example Request Body:

```
{
  "Scheduler_Name": "Daily Data Fetcher",
  "Source": "Internal API",
  "Date_Time_Scheduler_Ran": "2025-02-04T07:30:00Z",
  "Total_Records_Fetched": 1500,
  "Time_Taken": 120,
  "Status": "Success"
}
```

- **GET:** <http://127.0.0.1:8000/get-scheduler-logs/>
 - To select all records from the Scheduler_Logs table.

Action_History Table

- **Note:** This table will be populated when the Corporate_Actions table is updated. No need to insert data manually.
- **GET:** <http://127.0.0.1:8000/get-action-history/>
 - To select all records from the Action_History table.