# Detailed Pipeline for Procurement Chatbot System

## 1. Data Source

- **Dataset:** Public procurement data from Kaggle.
- **Storage:** MongoDB to store raw and processed data.

---

## 2. Backend EDA and Preprocessing

- **Data Loading:** Load raw procurement data from MongoDB using a Python script.
- **Exploratory Data Analysis (EDA):** Analyze data for trends, missing values, and inconsistencies.
- **Preprocessing:** Perform the following data cleaning steps:
    - Handle missing values.
    - Remove duplicates.
    - Optimize data types for better performance (e.g., converting categorical columns).
    - Standardize date formats and numeric columns.
    - Generate descriptive statistics to ensure data quality.

---

## 3. Chatbot Backend

- **Framework:** FastAPI.
- **Endpoints:**
    - `/api/v1/chat/submit/{chat_id}`:
        - Processes user queries, performs NLP, and retrieves relevant data.
    - `/api/v1/chat/history/{chat_id}`:
        - Retrieves user-specific chat history from MongoDB.
    - `/api/v1/process-data`:
        - Post-processes query results and returns structured output.

---

# 4. LLM-Powered Query Engine

- **Integration:** Utilize Groq  Open Source LLM(`llama-3.1-8b-instant`)
- and PandasQueryEngine for advanced query handling using  lama-index frame work.
- **Context Generation:** Summarize dataset structure and samples for LLM processing.
- **Query Execution:** Process user queries and provide insights from the procurement dataset.

---

# 5. Frontend Interface

- **Framework:** Streamlit.
- **Features:**
    1. Chat-based interface with user and bot avatars.
    2. Displays processed procurement data in a tabular format.
    3. Allows deletion of chat history.
    4. Displays error messages for failed interactions.
- **Workflow:**
    1. User enters a query.
    2. Query is sent to the backend via API.
    3. Processed results and LLM-generated insights are displayed.

---

# 6. Deployment and Maintenance

- **Deployment:**
    - Backend API: Hosted on FastAPI.
    - Frontend: Deployed as a Streamlit web app.

---

# Diagram Flow

1. **User Interface (Streamlit):** Users enter a query and receive chatbot responses along with procurement data visualizations.
2. **Chatbot API (FastAPI):** Interacts with MongoDB and the Groq LLM-based query engine.
3. **MongoDB:** Stores procurement data, chat history, and processed outputs.
4. **LLM Query Engine:** Executes advanced queries and generates human-readable insights.