**Assignment Title:**  
**AI-Powered Chatbot for Supplier and Product Information**

**Objective:**

Develop a chatbot that allows users to query a product and supplier database using natural language. The chatbot should interact with an open-source LLM and utilize LangGraph framework for agent workflows to fetch relevant information from a MySQL/PostgreSQL database and summarize the data using LLM.

**Assignment Deliverables:**

1. **Frontend (React):**
   * Create a responsive web interface to interact with the chatbot.
   * Provide a text input for users to enter their queries.
   * Display chatbot responses in a conversational format.
   * Provide a history of recent queries.
2. **Backend (Python & LangGraph):**
   * Implement a chatbot using LangGraph and Python.
   * Use LangGragh node to retrieve supplier and product information from MySQL/PostgreSQL.
   * Supplier data should be summarized using LLM
   * Use any open-source LLM (e.g., Hugging Face's GPT-2, GPT-3, or LLaMA 2) for summarization
3. **Database (MySQL/PostgreSQL):**
   * Design an efficient schema for storing:
     + Products (ID, name, brand, price, category, description, supplier ID).
     + Suppliers (ID, name, contact info, product categories offered).
   * Populate the database with sample data.
   * Implement queries to fetch relevant information based on chatbot requests.

**Functional Requirements:**

1. User inputs queries like:
   * "Show me all products under brand X."
   * "Which suppliers provide laptops?"
   * "Give me details of product ABC."
2. The chatbot should:
   * Fetch relevant data from the database.
   * enhance the response using context from LLM.
   * Return structured responses with relevant details.
3. The system should handle missing or incorrect queries gracefully.

**Technical Requirements:**

* **Backend:**
  + Python (FastAPI/Flask)
  + LangGraph for chatbot workflow
  + Open-source LLM (Hugging Face API, LLaMA 2, etc.)
* **Frontend:**
  + React (with Material UI or Tailwind CSS)
  + Axios for API calls
  + State management using Redux or Context API
* **Database:**
  + MySQL/PostgreSQL

**Evaluation Criteria:**

1. **Functionality:**
   * Completeness of chatbot interactions.
   * Accuracy of data retrieval and filtering.
2. **Code Quality:**
   * Clean, modular, and well-documented code.
3. **Scalability:**
   * Efficient query handling and performance.
4. **UI/UX:**
   * User-friendly design and responsiveness.

**Bonus Points (Optional Enhancements):**

* Implement authentication (JWT-based).
* Allow product comparisons.
* Add chatbot memory to recall user preferences.
* Include a simple analytics dashboard to track queries.

**Deadline:**  
3 days