AI-Powered Chatbot Project Documentation

Table of Contents

Introduction

This document provides a comprehensive overview of the Chatbot Project developed for the assignment. It includes details about the project's setup, database schema, API endpoints, functional requirements, and testing procedures.

Project Overview

The Chatbot Project is designed to provide users with information about products and suppliers using a natural language interface.
The chatbot can handle queries such as:
- "Show me all products under brand X."
- "Which suppliers provide laptops?"
- "Give me details of product ABC."

Installation and Setup

Prerequisites:
- Python 3.7+
- MySQL
- FastAPI
- Uvicorn
- PyTorch
- Additional Python packages (specified below)

Steps:
1. Clone the Repository:

git clone <repository-url>
cd chatbot-frontend/backend

2. Create a Virtual Environment and Activate it:
   python -m venv venv
   venv\Scripts\activate # For Windows
3. Install Required Packages:
   pip install fastapi uvicorn sqlalchemy pymysql transformers torch
4. Set Up the Database:
   - Create a MySQL database named chatbot_db.
   - Run the following SQL commands to create tables and insert sample data:

```
USE chatbot_db;

CREATE TABLE Suppliers (
   supplier_id INT AUTO_INCREMENT PRIMARY KEY,
   name VARCHAR(255) NOT NULL,
   contact_info VARCHAR(255),
   product_categories VARCHAR(255)
);

CREATE TABLE Products (
   product_id INT AUTO_INCREMENT PRIMARY KEY,
   name VARCHAR(255) NOT NULL,
   brand VARCHAR(255),
   price DECIMAL(10,2),
   category VARCHAR(255),
   description TEXT,
   supplier_id INT,
   FOREIGN KEY (supplier_id) REFERENCES Suppliers(supplier_id)
);

INSERT INTO Suppliers (name, contact_info, product_categories) VALUES
('TechCorp', 'techcorp@example.com, 123-456-7890', 'Electronics, Computers'),
('OfficeSuppliesCo', 'office@example.com, 098-765-4321', 'Furniture, Office Supplies'),
('BookWorld', 'books@example.com, 555-123-4567', 'Books, Stationery');

INSERT INTO Products (name, brand, price, category, description, supplier_id) VALUES
('Laptop', 'BrandX', 999.99, 'Electronics', 'High-performance laptop', 1),
('Office Chair', 'BrandY', 149.99, 'Furniture', 'Ergonomic office chair', 2),
('Notebook', 'BrandZ', 2.99, 'Stationery', '100-page notebook', 3),
('Smartphone', 'BrandX', 599.99, 'Electronics', 'Latest model smartphone', 1),
('Desk', 'BrandY', 249.99, 'Furniture', 'Spacious office desk', 2),
```

('Pen', 'BrandZ', 1.49, 'Stationery', 'Smooth-writing pen', 3);

5. Run the FastAPI Server:
   uvicorn main:app --reload

Database Schema

Suppliers Table:
- supplier_id: INT, Primary key, auto-increment
- name: VARCHAR(255), Supplier name
- contact_info: VARCHAR(255), Contact information
- product_categories: VARCHAR(255), Categories of products supplied

Products Table:
- product_id: INT, Primary key, auto-increment
- name: VARCHAR(255), Product name
- brand: VARCHAR(255), Product brand
- price: DECIMAL, Product price
- category: VARCHAR(255), Product category
- description: TEXT, Product description
- supplier_id: INT, Foreign key to suppliers table

API Endpoints

/chat (POST):
Description: Handles user queries and returns appropriate responses.

Request Example:
```
{
  "question": "Show me all products under brand BrandX."
}
```

Response Example:
```
{
  "answer": "Products under brand BrandX: ['Laptop', 'Smartphone']"
}
```

Functional Requirements

1. Show me all products under brand X:
   - Extracts the brand name from the query and retrieves products under that brand from

the database.
2. Which suppliers provide laptops?
  - Extracts the product category from the query and retrieves suppliers providing that category from the database.
3. Give me details of product ABC:
  - Extracts the product name from the query and retrieves product and supplier details from the database.

Testing and Debugging

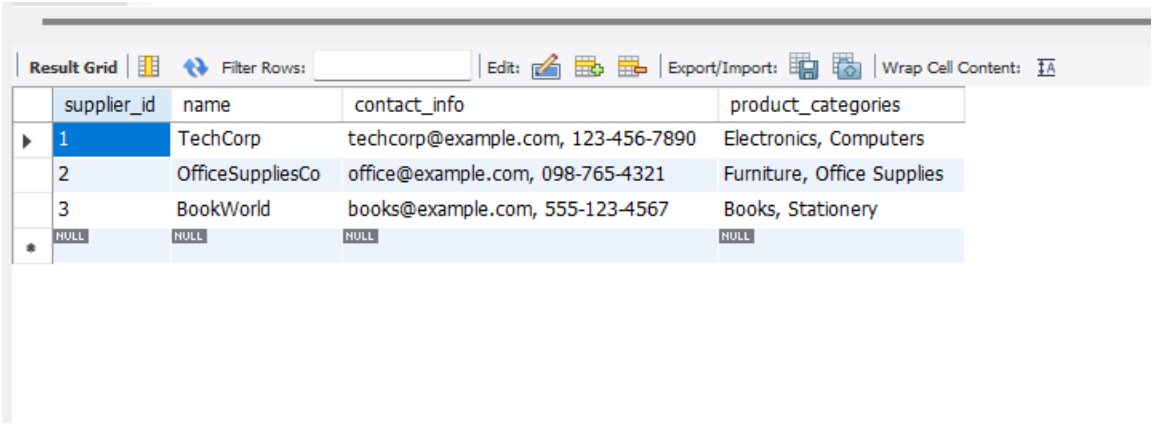
1. Debug Prints: Added debug print statements in main.py to track data processing.
2. Postman: Used Postman to send test requests and verify API responses.
3. Logs: Checked terminal logs for any errors and resolved them accordingly.

Conclusion

This project successfully implements a chatbot that can handle various user queries related to products and suppliers. The chatbot interacts with a database to fetch relevant information and provides accurate responses.

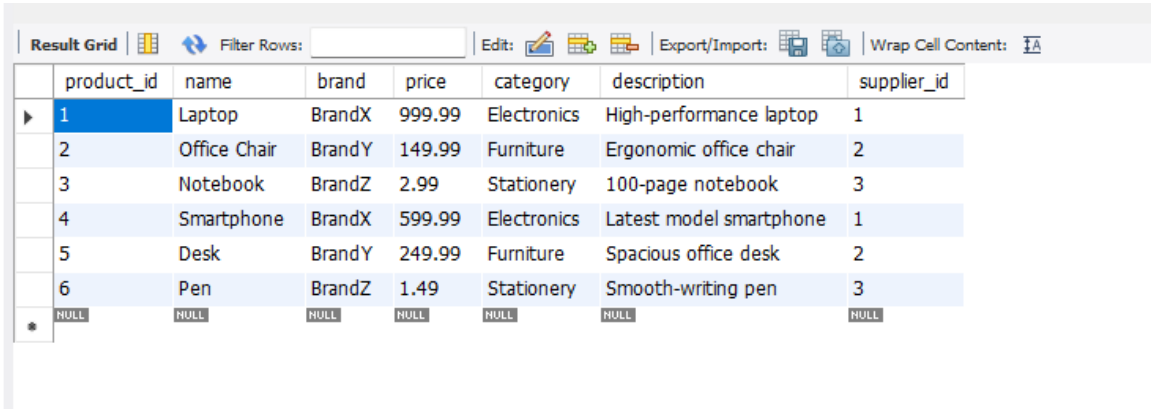
Screenshots

1. Screenshot of Database Setup:
   Description: Screenshot showing the database tables and sample data.



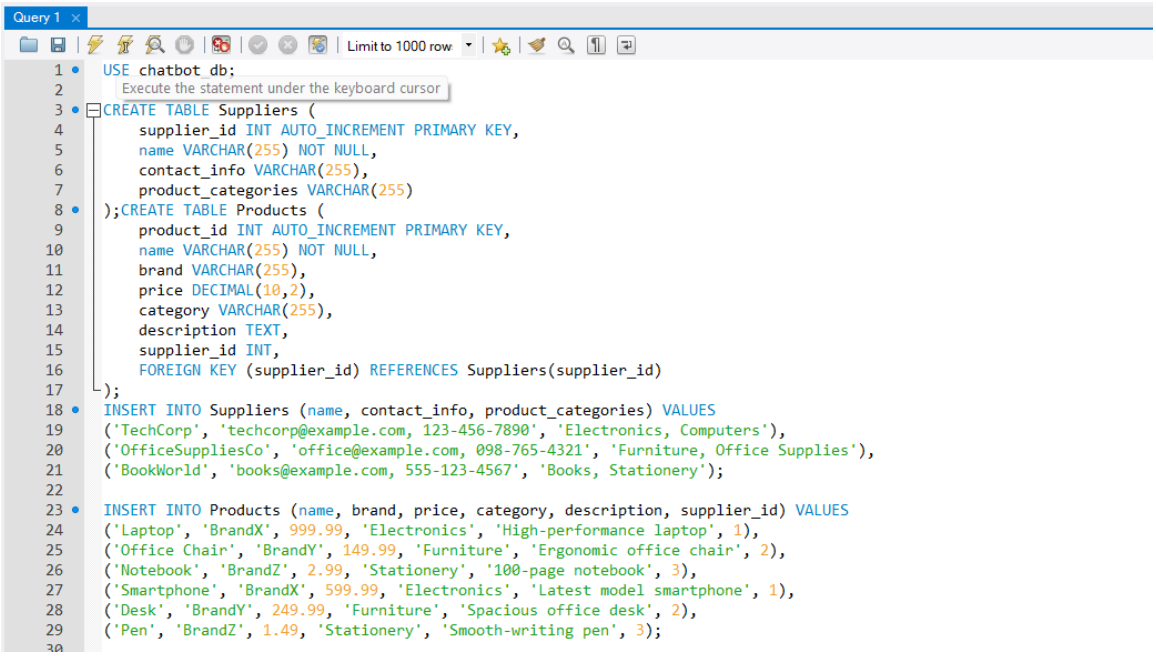
Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

supplier_id	name	contact_info	product_categories
1	TechCorp	techcorp@example.com, 123-456-7890	Electronics, Computers
2	OfficeSuppliesCo	office@example.com, 098-765-4321	Furniture, Office Supplies
3	BookWorld	books@example.com, 555-123-4567	Books, Stationery
NULL	NULL	NULL	NULL



Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

product_id	name	brand	price	category	description	supplier_id
1	Laptop	BrandX	999.99	Electronics	High-performance laptop	1
2	Office Chair	BrandY	149.99	Furniture	Ergonomic office chair	2
3	Notebook	BrandZ	2.99	Stationery	100-page notebook	3
4	Smartphone	BrandX	599.99	Electronics	Latest model smartphone	1
5	Desk	BrandY	249.99	Furniture	Spacious office desk	2
6	Pen	BrandZ	1.49	Stationery	Smooth-writing pen	3
NULL	NULL	NULL	NULL	NULL	NULL	NULL

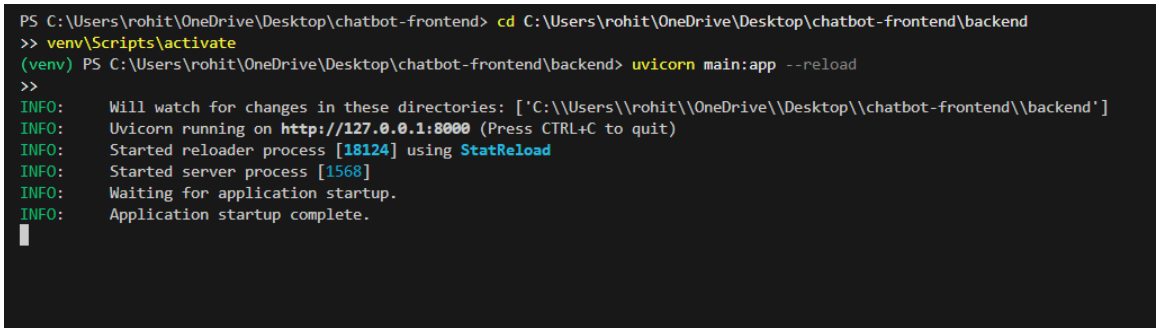


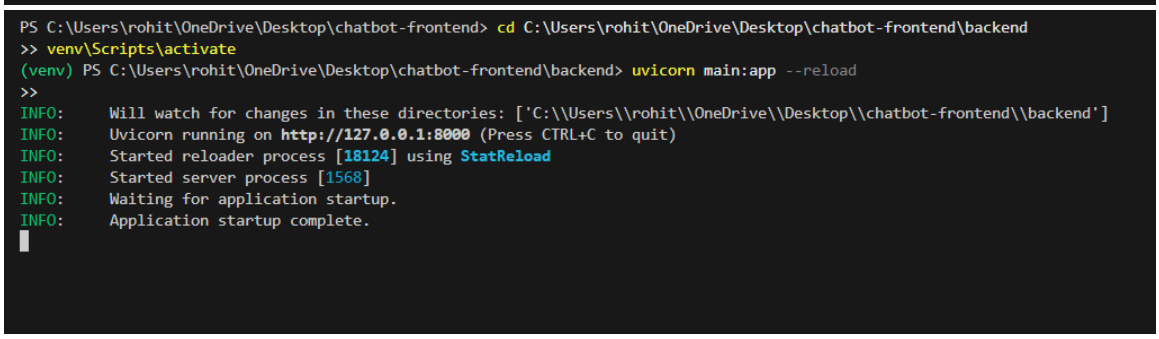
Query 1 | Limit to 1000 row

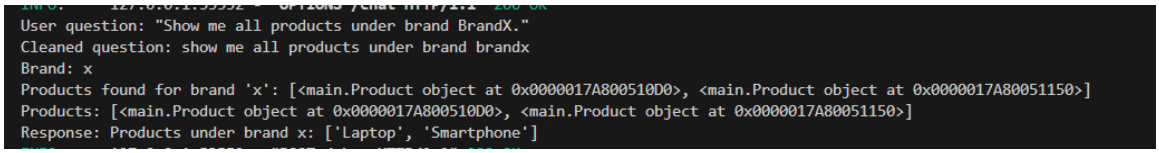
```
USE chatbot_db;
CREATE TABLE Suppliers (
    supplier_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    contact_info VARCHAR(255),
    product_categories VARCHAR(255)
);CREATE TABLE Products (
    product_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    brand VARCHAR(255),
    price DECIMAL(10,2),
    category VARCHAR(255),
    description TEXT,
    supplier_id INT,
    FOREIGN KEY (supplier_id) REFERENCES Suppliers(supplier_id)
);
INSERT INTO Suppliers (name, contact_info, product_categories) VALUES
('TechCorp', 'techcorp@example.com, 123-456-7890', 'Electronics, Computers'),
('OfficeSuppliesCo', 'office@example.com, 098-765-4321', 'Furniture, Office Supplies'),
('BookWorld', 'books@example.com, 555-123-4567', 'Books, Stationery');

INSERT INTO Products (name, brand, price, category, description, supplier_id) VALUES
('Laptop', 'BrandX', 999.99, 'Electronics', 'High-performance laptop', 1),
('Office Chair', 'BrandY', 149.99, 'Furniture', 'Ergonomic office chair', 2),
('Notebook', 'BrandZ', 2.99, 'Stationery', '100-page notebook', 3),
('Smartphone', 'BrandX', 599.99, 'Electronics', 'Latest model smartphone', 1),
('Desk', 'BrandY', 249.99, 'Furniture', 'Spacious office desk', 2),
('Pen', 'BrandZ', 1.49, 'Stationery', 'Smooth-writing pen', 3);
```

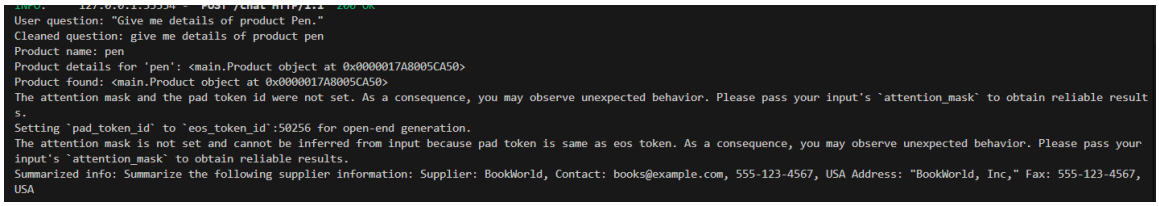
2. Screenshot of FastAPI Server Running:

Description: Screenshot showing the FastAPI server running in the terminal.









3. Screenshot of Frontend with Query Input:

Description: Screenshot showing the frontend interface with a query input box.