Case Study - Development of an E-commerce Sales Chatbot

In the competitive sphere of e-commerce, deploying advanced, interactive, and user-centric sales chatbots is essential for elevating customer experience and streamlining business operations. This case study presents the challenge of developing a comprehensive sales chatbot tailored for an e-commerce platform specialising in a specific product category (such as electronics, books, or textiles). The aim is to facilitate customer interactions from product search to purchase.

Objective:
Design and implement a sales chatbot that enhances the shopping experience by enabling efficient search, exploration, and purchase processes on an e-commerce platform. The deliverables include:
• The chatbot interface and logic.
• A simulated e-commerce server that processes user queries from the chatbot and returns relevant product data. This server should handle a mock inventory via RESTful interactions
(The backend data can be a mock e-commerce inventory)

Requirements:

1. User Interface / Front End:
• Develop a responsive user interface compatible with desktop, tablet, and mobile devices, utilizing modern JavaScript frameworks alongside HTML5 and CSS.
• Implement a login and authentication module to secure user sessions.
• Manage session continuity to maintain user state throughout interactions.
• Design a simple, intuitive chatbot interface with features such as conversation reset buttons and session tracking with timestamps.
• Ensure all chat interactions are stored effectively for later retrieval and analysis.

2. Backend
• Create an API-driven backend system using Python with frameworks like Flask or Django, capable of processing search queries and fetching relevant product data from a database.
• Populate a relational database management system (RDBMS) with at least 100 mock e-commerce product entries.

3. Technical Documentation:
• Document the entire process, including architecture , choice of tools/frameworks, and mock data creation should be documented
• Bonus if the candidate can include a section on potential challenges faced and how they were handled

4. Code Quality and Best Practices:
• The code should be clean, readable, and well-commented, adhering to industry standards for maintainability and scalability. Code should be implemented in python
• Ensure the codebase is modular and fault-tolerant, with clear separation of concerns and robust error handling.
• Provide rational for the choice of frameworks, libraries, and design patterns used in the project.

Evaluation Criteria:

1. UI User Experience:
• Creativity applied while visualising the products that are fetched without compromising the UX
• Innovative ways considered and applied to enable to customer interact , filter and explore the products

2. Technical Implementation:

• Quality of code, including readability, structure, and adherence to best practices for both the client (chatbot and server module)
• Modular architecture and fault tolerance capabilities of the architecture

3. Innovation and Problem-Solving:
• Creativity in approach and solutions to challenges encountered during the project.
• Ability to leverage advanced UI design / web development techniques catering to seamless customer product search experience.

4. Documentation and Presentation:
• Clarity and completeness of technical documentation, including code comments, project setup, and execution instructions.
• Effectiveness in communicating the project's objectives, methodology, results, and learnings.

Deliverables:
• A GitHub repository containing all source code, complete with a detailed README.md outlining project setup, execution instructions, and a comprehensive project summary.
• A detailed project report showcasing the technology stack used, sample queries, and the results obtained.
• A presentation to the recruitment panel, detailing the project approach, technologies utilized, and key learnings.