Software Requirements Specification (SRS) for **AI Chatbot for eCommerce Support**

---

1. Introduction

1.1 Purpose

The purpose of this document is to define the software requirements for an **AI Chatbot for eCommerce Support** that assists customers with order tracking, returns, product queries, and general inquiries by retrieving relevant information from the company's knowledge base.

1.2 Scope

This system will be a web-based AI chatbot integrated into an eCommerce platform. The chatbot will provide customers with real-time support, including:

- Order tracking and status updates
- Assistance with returns and refunds
- Product information and recommendations
- General inquiries (e.g., store policies, shipping details)

1.3 Definitions, Acronyms, and Abbreviations

- **AI**: Artificial Intelligence
- **Chatbot**: A conversational agent that interacts with users in real-time.
- **RAG**: Retrieval-Augmented Generation, used for real-time AI-based information retrieval.
- **FastAPI**: A high-performance backend framework for building APIs.
- **PostgreSQL**: A relational database for storing data.

1.4 References

- Next.js documentation
- FastAPI documentation
- Langchain RAG documentation
- PostgreSQL official guide
- eCommerce knowledge base API/documentation

---

2. Overall Description

2.1 Product Perspective

The AI Chatbot will be an integral part of the eCommerce platform and provide support for customer interactions. The chatbot will use **Langchain RAG** to retrieve relevant information from the company's knowledge base and **FastAPI** for backend communication. The web interface will be built using **Next.js**, and data such as order details and product information will be stored in **PostgreSQL**.

2.2 Product Features

1. **Order Tracking**: Allows customers to inquire about their order status.
2. **Returns and Refunds**: Provides guidance on returns, including generating return requests and tracking refunds.
3. **Product Queries**: Answers questions about product availability, specifications, and recommendations.
4. **General Inquiries**: Handles store policies, shipping options, and other frequently asked questions.
5. **Natural Language Processing**: Interprets and responds to user questions in natural language.
6. **Context-Aware Conversations**: Retains conversation context to handle follow-up questions.

2.3 User Characteristics

- **Customers**: Non-technical users interacting with the chatbot to resolve common issues.
- **Admin**: eCommerce staff responsible for managing the knowledge base and monitoring chatbot performance.

2.4 Constraints

- Must operate on existing eCommerce infrastructure.
- High concurrency: the chatbot should handle thousands of queries simultaneously.
- The system must ensure data privacy and security, especially for order tracking and payment information.

2.5 Assumptions and Dependencies

- The chatbot assumes that the knowledge base is regularly updated with relevant product and order information.
- Dependencies include access to order tracking APIs, shipping partner APIs, and return request handling.

---

3. Specific Requirements

3.1 Functional Requirements

3.1.1 Chatbot Interface

- **FR1**: The chatbot will be embedded in the eCommerce website, allowing customers to interact directly from the support section.
- **FR2**: The chatbot should respond to customer queries using natural language processing (NLP) to understand and interpret user input.

3.1.2 Order Tracking

- **FR3**: The chatbot must allow customers to enter their order ID or email and retrieve real-time order status using Langchain RAG from the knowledge base.
- **FR4**: The chatbot should handle multiple statuses, including shipped, in transit, and delivered.

3.1.3 Returns and Refunds

- **FR5**: The chatbot must guide customers through the return process, including generating return requests and providing refund timelines.
- **FR6**: The chatbot should retrieve return policy information and provide it to the customer.

3.1.4 Product Queries

- **FR7**: The chatbot must retrieve product details like availability, size, color, and price.
- **FR8**: The chatbot should offer product recommendations based on user input, including similar products or accessories.

3.1.5 General Inquiries

- **FR9**: The chatbot must provide information on store policies, such as shipping fees, delivery options, and refund policies.
- **FR10**: The chatbot should respond to questions about payment methods, coupons, and promotional offers.

3.1.6 Admin Dashboard

- **FR11**: Admins should be able to update the knowledge base with new product and policy information.
- **FR12**: Admins should have access to chatbot performance metrics, including query response times, customer satisfaction ratings, and frequently asked questions.

3.2 Non-Functional Requirements

3.2.1 Performance

- **NFR1**: The chatbot must handle a minimum of 1,000 concurrent queries.
- **NFR2**: Average response time should not exceed 2 seconds.

3.2.2 Scalability

- **NFR3**: The system should be horizontally scalable to accommodate an increasing number of users and data.

3.2.3 Reliability

- **NFR4**: The chatbot must have a 99.9% uptime to ensure it is available to customers at all times.

3.2.4 Security

- **NFR5**: The system must encrypt sensitive information such as order details and personal information.
- **NFR6**: Customer data must comply with GDPR and other relevant data privacy regulations.

3.2.5 Usability

- **NFR7**: The chatbot should have a simple and intuitive user interface with conversational prompts.
- **NFR8**: The language model must be trained to handle queries in different languages.

3.3 Database Requirements

3.3.1 PostgreSQL Integration

- **DR1**: The PostgreSQL database will store customer order data, product information, and chatbot conversation history.
- **DR2**: The database should support efficient querying of real-time data, ensuring fast response times for order tracking and product queries.

---

4. System Architecture

4.1 Web Interface (Next.js)

- The web application will be built with **Next.js**, providing a responsive UI for customers to interact with the chatbot.

4.2 Backend (FastAPI)

- The backend API will be built using **FastAPI**, handling user authentication, query processing, and integration with external services (e.g., shipping APIs).

4.3 AI Engine (Langchain RAG)

- **Langchain RAG** will serve as the AI retrieval engine, searching through the knowledge base and providing AI-generated responses to customer queries.

4.4 Database (PostgreSQL)

- All order details, user information, and chatbot interaction history will be stored and managed in **PostgreSQL**.

---

5. System Interfaces

5.1 API Interfaces

- **Order Tracking API**: Connects to shipping providers to retrieve real-time tracking updates.
- **Returns API**: Handles customer requests for returns and refunds.

5.2 User Interfaces

- **Customer UI**: The main chatbot interface embedded in the eCommerce website.
- **Admin Dashboard**: Allows eCommerce staff to manage the knowledge base and monitor chatbot performance.

---

6. Data Flow Diagram

1. **Customer Query → Chatbot (Next.js UI)**
2. **Chatbot → FastAPI Backend → Langchain RAG → Knowledge Base (Query Handling)**
3. **FastAPI Backend → PostgreSQL (Retrieve/Store Data)**
4. **FastAPI Backend → External API (Order Tracking/Returns)**

---

7. Testing Requirements

7.1 Functional Testing

- Test chatbot responses for common customer inquiries.
- Validate order tracking functionality with real data.
- Test chatbot behavior with edge cases (e.g., invalid order ID).

7.2 Performance Testing

- Load test the system to ensure it handles concurrent users efficiently.
- Measure average response time under high traffic.

7.3 Security Testing

- Perform vulnerability scans to ensure customer data is secure.
- Test compliance with GDPR and other relevant regulations.

---

8. Future Enhancements

1. **Voice Integration**: Add support for voice-based chatbot interactions.
2. **Multilingual Support**: Extend language support for non-English-speaking customers.
3. **Mobile App Integration**: Expand the chatbot to be part of the eCommerce mobile application.

This SRS provides a detailed plan for developing the **AI Chatbot for eCommerce Support**, ensuring that the system meets all functional, non-functional, and performance requirements.