Protean Support Assistant

Technical Documentation & Architecture Overview

Date: December 11, 2025

1. Project Overview

The Protean Support Assistant is a **Retrieval-Augmented Generation (RAG)** chatbot designed to provide accurate, professional answers regarding GRA (Ghana Revenue Authority) E-commerce compliance. Unlike a standard chatbot that might hallucinate facts, this system is grounded in a specific dataset (your FAQ document), ensuring high accuracy while maintaining a conversational tone.

2. How It Works (The Workflow)

The system follows a 4-step process to answer every user query:

- **Step 1: Ingestion**
  We read the source Word document (*GRA_Ecommerce_Chatbot_Intents_FAQs.docx*), extract the Questions and Answers, and store them in a structured JSON database.

- **Step 2: Vectorization (The Search Index)**
  We convert every FAQ question into a mathematical vector (a list of numbers) using **TF-IDF**. This creates a 'map' of our knowledge base.

- **Step 3: Retrieval (The Search)**
  When a user asks a question, we convert their text into a vector and compare it against our map using **Cosine Similarity**. This finds the most relevant FAQ entry mathematically.

- **Step 4: Generation (The AI)**
  We take the *correct answer* from our database and the *user's question*, and send them to **Google Gemini AI**. The AI then rewrites the answer to be friendly, professional, and conversational.

3. Deep Dive: The Search Mechanism

Algorithm: TF-IDF (Term Frequency-Inverse Document Frequency)

We chose TF-IDF over complex vector databases (like Pinecone or Chroma) for this specific use case.

Why?

- **Speed:** It is instant for datasets under 100,000 documents.

- **Keyword Precision:** In compliance/legal contexts, specific words (like 'VAT', 'TIN', 'Non-resident') matter more than general semantic meaning. TF-IDF is excellent at exact keyword matching.
- **Simplicity:** It runs locally without needing external servers or heavy API calls for embeddings.

4. Technology Stack & Rationale

Component	Technology	Why We Used It
Language	Python 3.10+	The standard for AI/ML development with vast library support.
Web Framework	Flask	Lightweight, fast, and perfect for microservices. We didn't need the overhead of
AI Model	Gemini 2.5 Flash	Google's latest efficient model. It provides high intelligence with very low latenc
Search Engine	Scikit-Learn	Industry-standard library for machine learning. We use its TfidfVectorizer and C
Frontend	Bootstrap 5 + JS	Ensures a professional, responsive UI that works on mobile and desktop withou
Data Parsing	python-docx	Allows us to directly read the raw MS Word file provided by the client.

5. Key Files Explained

- **rag_service.py:** The 'Brain'. This file contains the RAGChatbot class which handles loading data, training the search index, and talking to Gemini.
- **parse_faqs.py:** The 'Loader'. A utility script that converts the messy Word document into clean JSON data.
- **app.py:** The 'Server'. Handles HTTP requests from the browser and connects the frontend to the backend.
- **extracted_faqs.json:** The 'Database'. A local file acting as our knowledge base.