**Executive Summary**  
*Verba Support Agent — Final Project, Team 2377*

**Project Overview**  
Verba is a semantic troubleshooting assistant designed to provide fast and accurate support for smart home devices. Developed as part of a midterm assignment, the assistant processes natural language questions and returns mapped support suggestions using vector similarity techniques. The project was implemented entirely within Google Colab and features a live user interface deployed via Streamlit and Ngrok.

**Final Pipeline Architecture**  
The streamlined version demonstrated in class employs a focused, efficient retrieval system composed of:

* **Sentence Embedding:**  
  Queries are embedded using the MiniLM model (all-MiniLM-L6-v2) from the sentence-transformers library. This converts user input into dense semantic vectors.
* **FAISS Indexing:**  
  Embeddings are indexed using Facebook AI Similarity Search (faiss-cpu), enabling fast and scalable similarity matching against a predefined support corpus.
* **Response Mapping:**  
  Each vector corresponds to a human-readable troubleshooting solution, which is returned through a custom Python function.
* **Streamlit Web Interface:**  
  The assistant’s user-facing interface is built with Streamlit (app.py) and hosted publicly via pyngrok, enabling real-time interaction directly from Colab.

This pipeline ensures lightweight execution, modularity, and reproducibility for classroom demonstration and future development.

**Archived Explorations**  
Several advanced modules were prototyped as part of the early design phase. These were ultimately set aside to maintain simplicity and runtime stability in the final demo:

* **PDF Ingestion:**  
  Used pdfplumber to extract content from device manuals for dynamic corpus construction.
* **LangChain + ChromaDB:**  
  Employed text chunking with RecursiveCharacterTextSplitter and vector storage using HuggingFace embeddings inside a persistent Chroma database.
* **LLM-Powered QA Chain:**  
  Created a RetrievalQA system using HuggingFace’s flan-t5 and LangChain’s pipeline wrapper.
* **Feedback Loop & Evaluation Metrics:**  
  Simulated user interactions using manually crafted questions and keyword-based evaluation logic, including response accuracy and latency tracking.

While these components were not included in the final presentation, they represent promising directions for future iterations of Verba and showcase a broader understanding of retrieval-augmented design.

**Conclusion**  
Verba demonstrates how semantic AI techniques can be applied to real-world technical support systems. The project balances architectural clarity with performance, enabling fast query resolution and modular deployment. Its presentation in a Colab environment reflects both academic precision and practical usability