Intelligent Sri Lanka IRD Tax Intelligence & Compliance Assistant

1. Introduction and Problem Context

Tax-related information in Sri Lanka is published through official documents such as the Inland Revenue Act, tax guides, and circulars issued by the Inland Revenue Department (IRD). Although authoritative, these documents are often lengthy and legally complex, making it difficult for users to quickly locate and understand relevant information. Existing approaches such as keyword search or generic chatbots fail to provide reliable, context-aware answers, particularly in a compliance-sensitive domain like taxation. This project aims to address this gap by developing an intelligent tax assistant that delivers accurate, document-grounded responses with clear citations and disclaimers.

2. Overall System Design Approach

The system uses a modular architecture that separates the user interface, backend processing, and knowledge retrieval, making it easier to maintain and extend. Users interact through a web-based interface, while a FastAPI backend handles document ingestion, indexing, retrieval, and response generation. The system is powered by a Retrieval-Augmented Generation (RAG) pipeline, which retrieves relevant content from official tax documents to generate accurate, explainable, and source-backed answers.

3. Motivation for Using Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) was chosen over fine-tuning or prompt-only methods to keep the system accurate and easily updatable. Tax documents can be re-indexed without retraining the model, allowing the system to stay current. By grounding answers in retrieved document content, RAG reduces hallucinations and enables the use of citations, which is essential for accuracy and trust in a compliance-focused domain like taxation.

4. Document Ingestion and Indexing Design

The document ingestion pipeline processes official IRD PDF documents in a structured manner. Uploaded PDFs are stored in their original form to preserve traceability, after which text is extracted page by page along with relevant metadata. The text is then split into smaller chunks to improve retrieval efficiency. These chunks are converted into vector embeddings and stored in a vector database (ChromaDB), enabling fast and accurate semantic search during query processing.

5. Query Handling and Answer Generation

When a user submits a tax-related question, the FastAPI backend retrieves relevant document chunks from the vector database using semantic similarity search. A top-k retrieval strategy is applied to balance relevance and context. The retrieved content is passed to the RAG pipeline, where a language model generates an answer grounded in the provided documents. The response includes citations and a disclaimer indicating that the information is for guidance only and not professional tax advice.

6. Key Design Decisions and Trade-offs

Several design decisions were made to balance simplicity, performance, and flexibility. FastAPI was chosen for its performance, built-in validation, and automatic API documentation. Streamlit was used to enable rapid UI development and clear demonstration of system functionality. The system supports both local and cloud-based language models, allowing flexibility across different deployment environments. Explicit citations and disclaimers were included to ensure transparency and address ethical considerations in a compliance-sensitive domain like taxation.

7. Limitations and Future Enhancements

Although the system demonstrates a complete end-to-end RAG pipeline, there are several limitations. The current implementation assumes clean, text-based PDFs and does not handle scanned documents with poor text quality. Multilingual support for Sinhala and Tamil is also not yet implemented.Future enhancements could include incremental document updates, multilingual query support, confidence scoring for answers, and integration of authentication and access control. Deploying the system on a cloud platform with monitoring and logging would further improve its readiness for real-world use.

8. Conclusion

This project presents a practical and responsible approach to building an intelligent tax assistant using Retrieval-Augmented Generation. By grounding responses in official IRD documents, the system prioritizes accuracy, transparency, and maintainability. The modular design allows for future expansion while demonstrating strong engineering principles suitable for compliance-focused applications.