Architecture Roadmap: Document Ingestion & Analysis Feature (Phase III)

Project Lead: Mike Holland
System Architect: Gemini Pro
Objective: To extend the "Bob" AI Memory System with the capability to upload, process, and index text-based documents (e.g., PDF, DOCX, TXT), making their content available for retrieval in subsequent chat sessions.

1.0 Guiding Principles

- **Modularity:** This feature will be built as a new, distinct capability that integrates with, but does not disrupt, the existing, stable image and chat functionalities.
- **Separation of Concerns:** The process of ingesting a document will be handled by a dedicated n8n workflow. The main chat workflow will only be responsible for *retrieving* the indexed text, not processing the files.
- **Data Contract Adherence:** All new components must adhere to the established Master Data Contract, particularly in the use of project_id to link indexed content to its source project.

2.0 High-Level Architectural Design

The implementation will be broken down into three main components:

2.1 UI Modifications (The "Uploader")

The UI will be enhanced to support file uploads. This is more than just an "upload" button; it's a user-facing micro-application for managing the ingestion process.

- **New UI Component:** A new "Index Data" or "Upload Document" button will be added to the main UI. This will open a dedicated modal or view.
- **File Selection:** This new view will contain a file input element (<input type="file">) configured to accept common document types (.pdf, .docx, .txt).
- **Project Association:** The user must be able to select which existing project the new document should be associated with. This is critical for linking the data correctly.
- **API Call:** Upon submission, the UI will send the file data (likely via a multipart/form-data POST request) and the selected project_id to a new, dedicated n8n webhook.

2.2 New n8n Workflow: "Document Ingestion Pipeline" (The "Processor")

This will be a brand new, asynchronous workflow responsible for the entire file processing pipeline.

- **Webhook Trigger:** A new webhook will be created to receive the file and project_id from the UI.
- **File Storage (Optional but Recommended):** The first step should be to save the raw, uploaded file to a persistent storage location (e.g., a dedicated folder in Google Cloud Storage). This creates a backup and separates the ingestion process from the raw data.
- **Text Extraction:** This is the core of the new workflow. A new Code node or a specialized community node will be used to extract the raw text content from the uploaded file. This will require logic to handle different file types (e.g., using a library like pdf-parse for PDFs).
- **Integration with Existing RAG Pipeline:** Once the raw text is extracted, it will be passed to the **existing, proven logic** from your RAG Indexing Workflow. This includes:
    - The **Chunking** Code node.
    - The **Embedding** HTTPRequest node.
    - The final **Database Insert** Postgres node, which will save the chunks to the rag_store, correctly tagged with the project_id that was passed in at the beginning.
- **Response to UI:** The workflow will immediately respond to the UI with a "success, processing has begun" message, as the full indexing process may take some time. (Optional: A more advanced version could use a webhook to notify the UI upon completion).

2.3 Main Chat Workflow Modifications (The "Retriever")

The beauty of this architecture is that the Main Chat Workflow requires **almost no changes.**

- Because the new "Document Ingestion Pipeline" saves its output to the same rag_store table, using the same project_id as all other memory types, the existing Retrieve RAG Chunks node will **automatically** be able to find and retrieve this new information. Its WHERE project_id = $1 query is universal.

3.0 Next Steps

This document serves as the high-level blueprint. The next step is to hand this to your new Chief Engineer.

Your prompt to the copilot:

"Here is the architectural roadmap for our new Document Ingestion feature. Please review it and provide a detailed, step-by-step implementation plan, including the proposed UI changes and the node-by-node structure of the new 'Document Ingestion Pipeline' n8n workflow."