**Self Healing Architecture Overview**

**Phase 1: Document Processing Pipeline (Offline/Independent)**

This phase runs independently to prepare documents for retrieval:

* **Documents → Chunking**: Raw documents are split into smaller, manageable segments
* **Chunking → Preprocessing/Metadata Extraction**: Each chunk is enriched with relevant metadata
* **Preprocessing → Embedding Model**: Text chunks are converted to numerical vector representations
* **Embedding Model → Vector DB**: Vectors are stored in a specialized database for fast similarity search

**Note: Phase 1 operates independently and can run offline. Documents can be processed and indexed at any time without affecting query operations.**

**Phase 2: Query Processing & Response Generation (Runtime/Independent)**

This phase activates only when a user submits a query:

1. **User → Query**: User input is received by the system
2. **Query → Query Routing**: The query is analyzed to determine the best processing path
3. **Query Routing → AI Agent**: An intelligent agent processes and enhances the query
4. **AI Agent → Embedding Model**: The query is converted to a vector using the same embedding model
5. **Embedding Model → Vector DB**: The query embedding vector is sent to the database for similarity search
6. **Vector DB → Retrieval → Context**: Relevant document chunks are retrieved based on vector similarity
7. **Context + Query → Augmentation**: Original query is enhanced with retrieved context
8. **Augmentation → LLM**: The augmented query is sent to the Large Language Model
9. **LLM → Generation → User**: The final response is generated and delivered to the user

**Note: Phase 2 operates independently at runtime. It only requires the Vector DB to be populated from Phase 1.**

**Key Architecture Benefits**

* **Independent Phases**: Phase 1 (indexing) and Phase 2 (querying) operate completely independently
* **Asynchronous Processing**: Documents can be indexed without interrupting query operations
* **Consistent Embeddings**: Same model used for both document and query vectorization
* **Context-Aware Generation**: LLM receives relevant context for accurate responses
* **Scalable Design**: Each phase can be scaled independently based on workload

