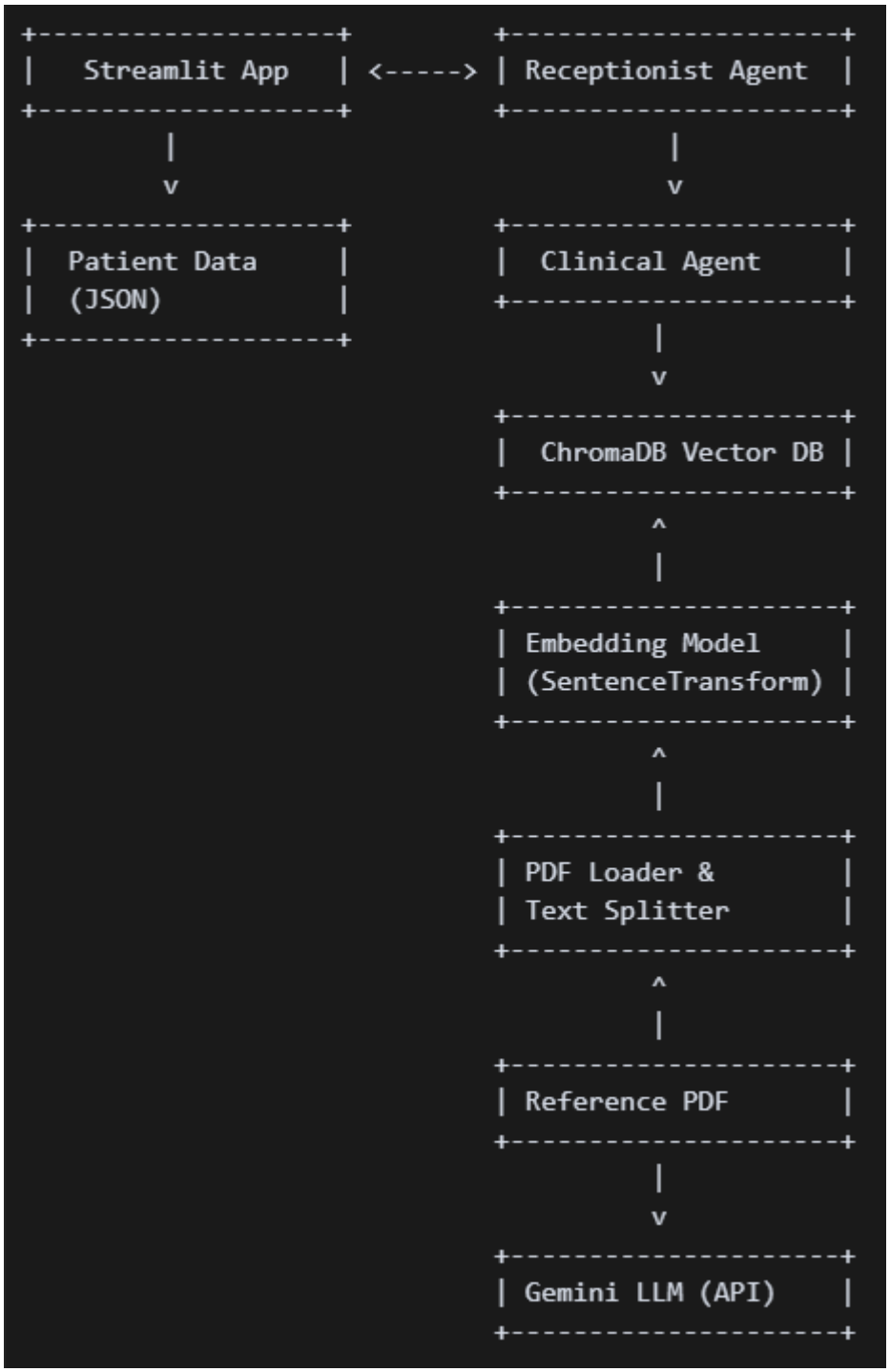
Post-Discharge Medical AI Assistant

1. **Architecture Overview**



2. Component Justification

A. Streamlit App (Frontend)

Justification: Provides a simple, interactive web interface for patients and clinicians. Streamlit enables rapid prototyping and easy deployment without complex frontend development.

B. Receptionist Agent

Justification: Handles patient engagement, follow-up prompts, and initial triage. This separation allows for a more human-like, guided experience and keeps the clinical agent focused on medical Q&A.

C. Patient Data (JSON)

Justification: Using JSON for patient data is lightweight, easy to edit, and integrates well with Python. It's suitable for prototyping and can be replaced with a database or EHR integration in production.

D. Clinical Agent

Justification: Centralizes the logic for answering medical questions. It first attempts to answer using trusted reference material (RAG), and only falls back to the LLM if needed, ensuring reliability and safety.

E. ChromaDB Vector Database

Justification: Stores vector embeddings of reference material for fast, semantic search. This enables the system to retrieve relevant information from large, unstructured documents efficiently.

F. Embedding Model (Sentence Transformers)

Justification: Converts text into vector representations for semantic search. Sentence Transformers are efficient and provide high-quality embeddings for medical and general text.

G. PDF Loader & Text Splitter

Justification: Automates the extraction and chunking of reference material, making it easy to update or expand the knowledge base with new documents.

H. Reference PDF

Justification: Serves as the authoritative source of clinical knowledge, ensuring that answers are grounded in real, trusted material.

I. Gemini LLM (API)

Justification: Acts as a fallback for questions not covered in the reference material, ensuring the assistant can always provide a helpful response. The system clearly warns users when LLM-generated content is used.

3. Design Principles

- **Modularity:** Each component (agents, data, embeddings, frontend) is separated, making the system easy to maintain, extend, and test.
- **Reliability:** Answers are grounded in reference material whenever possible, with LLM fallback only when necessary.
- **Extensibility:** New agents, data sources, or reference materials can be added with minimal changes.
- **User-Centric:** The interface and workflow are designed for ease of use by patients and clinicians.
- **Transparency & Safety:** The system logs all interactions and displays clear disclaimers when using LLM-generated content.

4. Scalability & Future-Proofing

- **Reference Material:** Easily swap or add new PDFs for other specialties.
- **Data Sources:** Upgrade from JSON to a database or EHR integration as needed.
- **Agents:** Add more specialized agents (e.g., pharmacist, social worker) for broader support.
- **LLM Flexibility:** Swap Gemini for another LLM or local model as requirements evolve.

5. Conclusion

This architecture balances rapid development, reliability, and extensibility. By combining semantic search over trusted references with LLM fallback, it delivers safe, context-aware support for post-discharge patients, and is well-positioned for future enhancements.