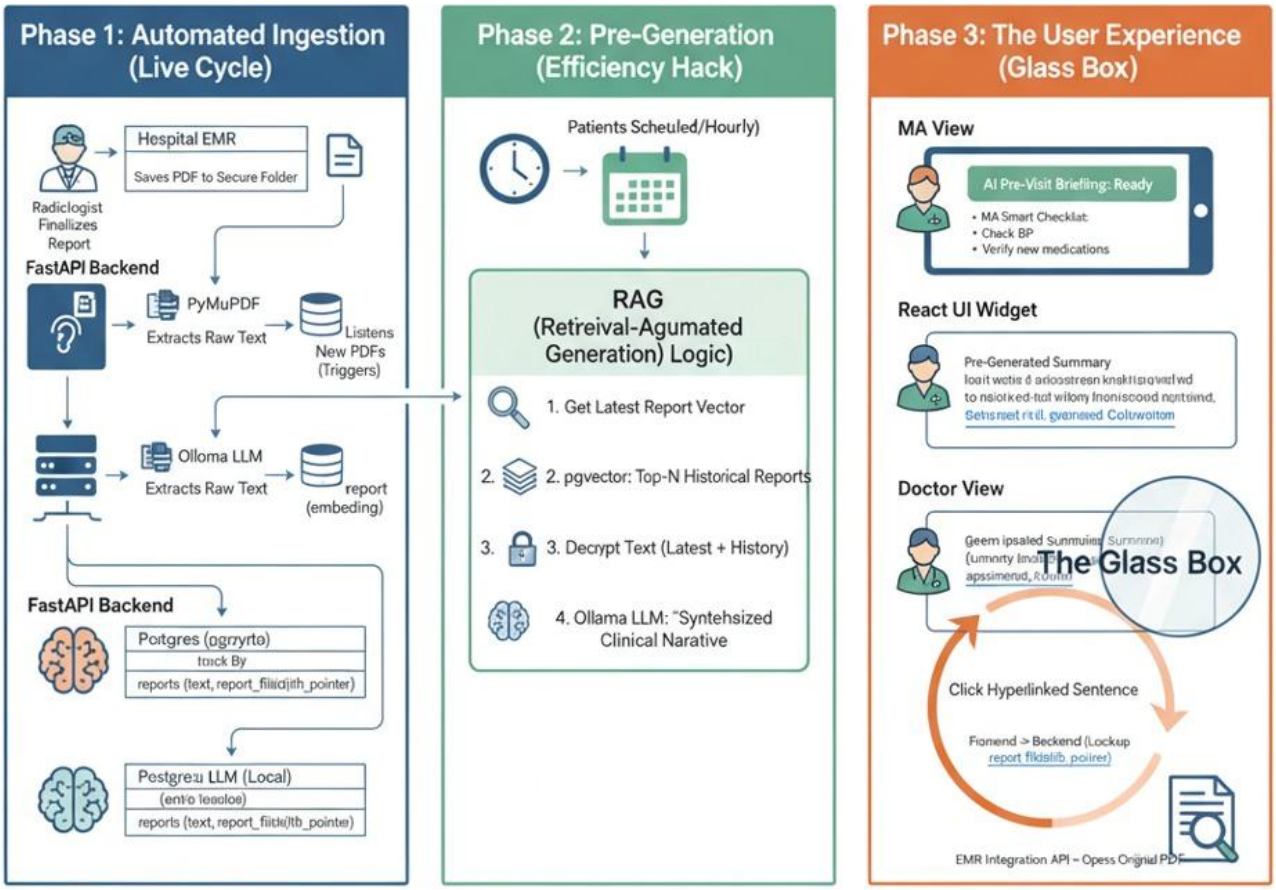
Phase 1: Automated Ingestion (Live Cycle)
Hospital EMR
Saves PDF to Secure Folder
Radiologist Finalizes Report
FastAPI Backend
PyMuPDF
Extracts Raw Text
Listens New PDFs (Triggers)
Olloma LLM
Extracts Raw Text
report (embeding)
FastAPI Backend
Postgres (pgvector)
reports (text, report_filepath_pointer)
Postgres LLM (Local)
reports (text, report_filepath_pointer)
Phase 2: Pre-Generation (Efficiency Hack)
Patients Scheuled/Hourly)
RAG (Retreival-Agumated Generation) Logic)
1. Get Latest Report Vector
2. 2. pgvector: Top-N Historical Reports
3. 3. Decrypt Text (Latest + History)
4. Ollama LLM: "Syntehsized Clinical Narative
Phase 3: The User Experience (Glass Box)
MA View
AI Pre-Visit Briefing: Ready
• MA Smart Checklist
• Check BP
• Verify new medications
React UI Widget
Pre-Generated Summary
Doctor View
The Glass Box
Click Hyperlinked Sentence
Frontend -> Backend (Lookup report filepath pointer)
EMR Integration API – Opens Original PDF