Product Requirements Specification (PRS)

1. Product Overview

- **Product Name:** SummAID
- **One-line description:** An on-premise AI assistant that synthesizes complex patient medical history from various reports into a concise, verifiable pre-visit summary for clinical teams.
- **Target Users:**
  - **Primary User:** Medical Assistant (for chart preparation).
  - **Beneficiary:** Doctor (GP/Specialist) (for reviewing patient context).
  - **Buyer:** Hospital Administrators / IT / Clinical Informatics.
- **Value Proposition:** Dramatically reduces clinician prep time, decreases cognitive load/burnout, and improves patient safety by providing accurate, synthesized, and verifiable patient context directly within the clinical workflow.

2. Problem Statement

- **Current gap or pain point:** Clinicians spend excessive time manually reviewing fragmented and lengthy patient data across multiple EMR reports (Radiology, Pathology, Labs, Notes) before appointments. This is inefficient, contributes to burnout, and increases the risk of missing critical information.
- **Why existing solutions are insufficient:** Standard EMRs display data but lack deep synthesis capabilities. Existing summarization tools are often cloud-based (raising privacy concerns), lack cross-report synthesis, or are "black boxes" lacking verifiable citations, hindering clinical trust.

3. Objectives

- **High-level goals of the product:**
  - Reduce pre-visit chart review time for clinical staff significantly.
  - Improve clinician confidence by providing accurate, synthesized patient context.
  - Enhance patient safety by minimizing the risk of missed critical findings.
  - Ensure data privacy and security through an on-premise architecture.
  - Build clinical trust through verifiable ("Glass Box") citations.
- **Success criteria:**
  - Measurable reduction in average chart prep time during pilot study (e.g., >50%). ○ High usability and satisfaction scores from MAs and Doctors in pilot feedback (>8/10).
  - Demonstrably high accuracy in summaries (e.g., >99% recall of critical findings) validated by clinicians.
  - Successful deployment and stable operation within a test hospital environment.

4. Scope

- **In Scope:**
  - Automated processing of historical and new text-based medical reports (PDF primarily).
  - Secure, encrypted storage of extracted text and generated vector embeddings within an on-premise PostgreSQL database.
  - AI-powered retrieval of relevant historical reports based on semantic similarity to the latest report (pg vector).
  - Generation of synthesized summaries using an on-premise LLM (Ollama/Llama 3).
  - Generation of verifiable, clickable citations linking summary statements to source text.
  - Integration with EMR via FHIR API (read-only for reports initially).
  - Basic UI ("Fake EMR" for demo, integrated widget for pilot) for MAs/Doctors.
  - Automated "MA Smart Checklist" generation based on summary.
- **Out of Scope:**
  - Processing image-only or non-text report formats (initially).
  - Direct EMR data writing capabilities (initially read-only).
  - Real-time summarization during patient encounters.
  - Making clinical diagnoses or treatment recommendations (decision support, not decision making).
  - User accounts/authentication (will rely on EMR's SSO in production).
  - Cloud-based deployment or APIs.
  - Billing/coding features (Potential Phase 2).
  - Patient-facing summaries (Potential Phase 2).

5. Core (Functional) Features

- **Feature 1: Automated Report Processing & Storage**
  - **User Interaction Flow:** System automatically detects new PDF report in designated EMR folder -> Extracts text (PyMuPDF) -> Encrypts text (pg crypto) -> Generates vector embedding (Ollama) -> Stores encrypted text, vector (pg vector), and metadata in PostgreSQL.
  - **AI/ML Role:** Text Extraction, Embedding Generation.
  - **Acceptance Criteria:** New reports are processed and stored correctly within 5 minutes of appearing in the monitored folder; data is verified as encrypted in the DB.
- **Feature 2: Relevant Context Retrieval**
  - **User Interaction Flow:** System identifies latest report for an upcoming appointment -> Generates/retrieves its vector -> Queries pg vector index -> Returns top N most semantically similar historical report_ids for that patient. ○ **AI/ML Role:** Vector Similarity Search (Retrieval).

○ **Acceptance Criteria:** Retrieval completes in <1 second; retrieved reports are demonstrably relevant to the latest report based on clinical review. ● **Feature 3: Synthesized Summary Generation**

  - **User Interaction Flow:** System retrieves decrypted text for latest report + N relevant reports -> Constructs prompt for LLM -> Sends combined text to local Ollama -> Receives synthesized summary text.
  - **AI/ML Role:** Text Generation (Summarization/Synthesis).
  - **Acceptance Criteria:** Summary is generated in <10 seconds; summary accurately reflects key findings and connections across input reports; summary is concise and clinically useful.

- **Feature 4: Verifiable Citation Generation ("Glass Box")**
  - **User Interaction Flow:** During/after summary generation, system maps summary sentences back to specific sentences/locations in the original source reports -> Embeds these links/pointers within the summary output.
  - **AI/ML Role:** Information Extraction / Mapping (Potentially requires advanced LLM prompting or separate logic).
  - **Acceptance Criteria:** >95% of summary sentences have accurate, clickable citations; clicking a citation correctly highlights/displays the source text.
- **Feature 5: MA Smart Checklist Generation**
  - **User Interaction Flow:** System analyzes generated summary for actionable items or missing information -> Generates a short list of tasks for the MA -> Displays checklist in the UI.
  - **AI/ML Role:** Information Extraction / Rule-based Task Generation.
  - **Acceptance Criteria:** Checklist items are relevant to the summary; tasks are clear and actionable for an MA.

6. Data Requirements

- **Data Sources:** Unstructured text from clinical reports (Radiology, Pathology initially) stored as PDFs within the hospital's EMR/PACS system. Patient scheduling data (to identify upcoming appointments).
- **Data Size & Quality Expectations:** Needs to handle potentially thousands of historical reports per patient, and potentially millions across a hospital. Assumes source reports are reasonably well-structured text PDFs (not scanned images needing OCR initially). Expects standard medical terminology.
- **Privacy / Compliance:** HIPAA (US) / DPDPA (India) / relevant local regulations are paramount. All processing must occur on-premise. Data at rest must be encrypted (pg crypto). Strict access controls (via EMR SSO) and audit logs are required. No PHI should ever leave the hospital network.

7. System Behavior

- **Inputs:** New PDF reports appearing in a monitored folder, patient appointment schedule, user clicks (MA/Doctor interaction with UI).

- **Outputs:** Synthesized summary text with embedded citations, MA checklist, status updates in UI, detailed audit logs.
- **Performance (Non-Functional) Expectations:**
    - **Latency:** Summary retrieval/display < 10 seconds. Vector search < 1 second.
    - **Accuracy:** Clinical accuracy of summary validated as extremely high (>99% critical finding recall). Citation accuracy >95%.
    - **Reliability:** System should have high uptime; automated processing should be robust to common PDF variations. Graceful failure if summarization fails (e.g., "Summary unavailable").

8. Constraints

- **Technical:**
    - **Deployment:** Must be deployable entirely on-premise using Docker containers.
    - **Hardware:** Requires server infrastructure provided by the hospital, including at least one compatible NVIDIA GPU for the LLM.
    - **Integration:** Must integrate with EMR, initially via FHIR API for read access.
    - **LLM:** Must use locally runnable open-source models via Ollama. No cloud AI APIs.
- **Regulatory:** Strict adherence to HIPAA/DPDPA/local data privacy laws.
- **Ethical:**
    - **Bias:** Monitor and mitigate potential biases in summarization (though less risk than diagnostic AI).
    - **Explainability:** "Glass Box" citations are key to providing explainability.
    - **Responsibility:** Tool is for decision support, clinical user retains final responsibility. Output should clearly state it's AI-generated.

9. Metrics & Evaluation

- **Key evaluation metrics:**
    - **Clinical Accuracy:** Recall/Precision/F1 score for critical findings in summaries (manual review by clinicians).
    - **Performance:** End-to-end summary generation/display latency (seconds).
    - **Workflow Impact:** Reduction in MA/Doctor chart prep time (minutes per patient, measured in pilot).
    - **User Satisfaction:** MA & Doctor feedback scores (e.g., Likert scale 1-10).
    - **Citation Accuracy:** Percentage of summary sentences with correct source links.
- **Target thresholds:**
    - **Must-have:** Latency < 10s, Critical Finding Recall >99%, User Satisfaction > 7/10.
    - **Nice-to-have:** Latency < 5s, User Satisfaction > 9/10, measurable ROI demonstration.

10. Implementation Notes

- **Deployment plan:** Docker containers deployed on hospital-provided on-premise server(s). Integration via FHIR adapter module within the backend.
- **Monitoring requirements:** Basic application logging (errors, uptime), detailed audit logs for data access, monitoring of LLM performance/resource usage, potentially monitoring for "hallucinations" or repetitive outputs.

11. Timeline & Priorities

- **Phase 1 (MVP - IA2):**
  - **Deliverable:** Functional "Canned Demo" (as detailed previously).
  - **Focus:** Core backend (DB setup, seeding, basic API), core frontend (Fake EMR UI), basic summarization & citation logic using local LLM & sample data. Prove technical feasibility.
- **Phase 2 (SEE / Pre-Pilot):**
  - **Deliverable:** Enhanced demo ready for pilot integration.
  - **Focus:** Refine synthesis, improve citation accuracy, build FHIR adapter, enhance UI based on feedback, implement MA checklist.

12. Open Questions

- **Decisions pending:**
  - Specific LLM fine-tuning strategy (if needed beyond base Llama 3).
  - Exact number (N) of relevant reports to retrieve via pgvector.
  - Detailed UI design for MA checklist and doctor's summary view within EMR context.
  - Specific GPU requirements for target performance.
- **Assumptions to validate:**
  - That MAs/Nurses see value in the automated checklist.
  - That doctors will trust and rely on the summaries given the "Glass Box" citations.
  - That hospitals are willing/able to provide the necessary on-premise GPU hardware for a pilot.
  - That FHIR APIs provide sufficient access to necessary report data
  - That PyMuPDF can reliably extract text from the hospital's typical PDF report formats.