NeuroAssist AI – Detailed Software Requirements Document (SRD)

Version: 2.0 (Verified Implementation) **Date**: 2026-01-10 **Source**: Derived from NeuroAssist MAX PRD v4.0

> ***Purpose***: *This document translates the Product Requirements into strict, testable technical requirements for engineering and QA. All statements use the standard "The system shall..." format.*

---

1. System Architecture & Deployment Requirements

- **SRD-1.1**: The system **shall** be architected as a set of Docker containers: Frontend (React/Nginx), Backend (FastAPI/Uvicorn), and Database (PostgreSQL).
- **SRD-1.2**: The backend service **shall** be stateless to allow horizontal scaling behind a load balancer.
- **SRD-1.3**: The system **shall** be deployable on a local developer machine using `docker-compose` .
- **SRD-1.4**: The system **shall** be compatible with cloud-managed PostgreSQL services (e.g., AWS RDS, GCP Cloud SQL).
- **SRD-1.5**: All inter-service and client-server communication **shall** be encrypted using TLS 1.2 or higher (HTTPS).

2. User Roles & Access Control Requirements

- **SRD-2.1**: The system **shall** enforce the following distinct user roles: `DOCTOR` , `FRONT_DESK` , `ADMIN` , `PATIENT` .
- **SRD-2.2**: The system **shall** implement Role-Based Access Control (RBAC) at the API endpoint level using a dependency injection pattern (e.g., `RoleChecker`).
- **SRD-2.3**: Front Desk users **shall not** have permission to view clinical transcripts, audio files, or SOAP notes (HIPAA Minimum Necessary Rule).
- **SRD-2.4**: Doctors **shall** have full read/write access to consultations assigned to them or unassigned in the queue.
- **SRD-2.5**: Patients **shall** only have read access to their own historical data.

3. Audio Capture & Storage Requirements

- **SRD-3.1**: The frontend **shall** provide a single, accessible UI control to toggle recording Start/Stop.
- **SRD-3.2**: The frontend **shall** render a real-time visual waveform of the audio input to confirm microphone activity.
- **SRD-3.3**: The system **shall** automatically stop recording if the duration exceeds 15 minutes.
- **SRD-3.4**: Recorded audio files **shall** be stored in an encrypted file system or object store (AES-256).
- **SRD-3.5**: The system **shall** provide a "Upload Audio" fallback mechanism accepting `.mp3` , `.wav` , and `.m4a` files up to 50MB.
- **SRD-3.6**: Every audio file **shall** be linked via foreign key to a specific `consultation_id` .

4. Transcription Requirements

- **SRD-4.1**: The system **shall** utilize the AssemblyAI API for asynchronous speech-to-text processing.
- **SRD-4.2**: The transcription request **shall** enable Speaker Diarization to distinguish between "Doctor" and "Patient".
- **SRD-4.3**: The transcription request **shall** enable PII Redaction for `medical_process` and `us_social_security_number` entities.

- **SRD-4.4**: The text transcript **shall** be persisted in the `audio_files` database table.
- **SRD-4.5**: The "Transcript Tab" **shall** display the text separated by speaker labels (e.g., "Speaker A", "Speaker B").
- **SRD-4.6**: The system **shall** provide a mechanism to "Reset" or "Clear" a transcript if the doctor deems it inaccurate.

5. Transcript Visibility & Privacy Requirements

- **SRD-5.1**: The Transcript UI component **shall** be conditionally rendered only for users with `DOCTOR` or `ADMIN` roles.
- **SRD-5.2**: API endpoints returning consultation details **shall** omit transcript fields when accessed by `FRONT_DESK` users.

6. SOAP Note Generation Requirements

- **SRD-6.1**: The system **shall** use the Google Gemini 2.5 Flash model for generating clinical notes.
- **SRD-6.2**: The AI prompt **shall** enforce a strict JSON output schema containing `subjective`, `objective`, `assessment`, and `plan`.
- **SRD-6.3**: The system **shall** allow the Doctor to manually edit any section of the generated SOAP note.
- **SRD-6.4**: The system **shall** support a "Regenerate" action that re-submits the transcript to the LLM.
- **SRD-6.5**: SOAP generation **shall** fail gracefully and return an error if no transcript exists.

7. Medical Terminology / Word Bank Requirements

- **SRD-7.1**: The transcription service **shall** accept a list of custom vocabulary ("word_boost") to improve accuracy.
- **SRD-7.2**: The current configuration **shall** include: "Levetiracetam", "Sumatriptan", "Ataxia", "Donepezil".
- **SRD-7.3**: The architecture **shall** support loading these terms from a database table (`medical_terms`) in the future.

8. Triage & Queue Management Requirements

- **SRD-8.1**: The `TriageService` **shall** calculate a risk score (0-100) based on keyword checking of the patient's pre-visit transcript.
- **SRD-8.2**: The system **shall** categorize urgency as:
  - **CRITICAL** (Score ≥ 95): Keywords like "Stroke", "Suicide".
  - **HIGH** (Score ≥ 75): Keywords like "Fainting", "Severe Pain".
  - **MODERATE** (Score ≥ 50).
  - **LOW** (Score < 50).
- **SRD-8.3**: The Front Desk Queue **shall** default to sorting patients by Urgency Score (Descending), then Wait Time (Descending).
- **SRD-8.4**: Patients with `CRITICAL` status **shall** be displayed with a visual Red Badge indicator.
- **SRD-8.5**: The Triage evaluation **shall** complete within 1 second of transcript availability.

9. Front Desk Functional Requirements

- **SRD-9.1**: The Front Desk Dashboard **shall** auto-refresh the patient queue status every 15-30 seconds.
- **SRD-9.2**: The Dashboard **shall** display Patient Name, Wait Time, and Triage Priority.

- **SRD-9.3**: Clicking a patient row **shall** open a "Check-in" view that excludes clinical data (Audio/Transcript).
- **SRD-9.4**: The system **shall** allow Front Desk users to assign a specific Doctor to a patient appointment.

10. Admin Functional Requirements

- **SRD-10.1**: Admin users **shall** have a UI to provision new User accounts (`DOCTOR` , `FRONT_DESK`).
- **SRD-10.2**: The system **shall** expose a health check endpoint `/health` for monitoring service status.
- **SRD-10.3**: Use of AI services **shall** be logged in the `ai_logs` table for cost and error tracking.

11. Error Handling & Fallback Requirements

- **SRD-11.1**: If the browser denies microphone access, the UI **shall** display a modal instructions guiding the user to enable permissions.
- **SRD-11.2**: If the LLM service returns a 5xx error, the system **shall** retry the request up to 3 times with exponential backoff.
- **SRD-11.3**: If AI generation fails ultimately, the system **shall** default to an empty SOAP form for manual entry.

12. Data & Database Requirements

- **SRD-12.1**: The system **shall** use PostgreSQL as the primary persistent store.
- **SRD-12.2**: The `consultations` table **shall** have a B-Tree index on `urgency_score` to optimize queue sorting.
- **SRD-12.3**: All critical entities (`Users` , `Consultations` , `SOAPNotes`) **shall** have `created_at` and `updated_at` timestamps.
- **SRD-12.4**: The system **shall** maintain an `audit_logs` table recording: Actor ID, Action Type, Target Resource, and Timestamp.

13. Security & Compliance Requirements

- **SRD-13.1**: API Authentication **shall** utilize stateless JSON Web Tokens (JWT).
- **SRD-13.2**: Access Tokens **shall** have a time-to-live (TTL) of 15 minutes or less.
- **SRD-13.3**: Patient Health Information (PHI) **shall** be encrypted at rest in the database and file storage.
- **SRD-13.4**: The system **shall** log detailed audit trails sufficient for HIPAA compliance reviews.

14. Performance Requirements

- **SRD-14.1**: The Login API endpoint **shall** respond in under 500ms (95th percentile).
- **SRD-14.2**: The Audio Upload endpoint **shall** support files up to 50MB with a timeout no less than 60 seconds.
- **SRD-14.3**: SOAP Note generation **shall** complete within 10 seconds under normal load.
- **SRD-14.4**: Triage scoring **shall** process within 300ms of text availability.

15. Testing & Validation Requirements

- **SRD-15.1**: Backend codebase **shall** maintain unit test coverage for all Service Logic (Triage, Auth, Upload).
- **SRD-15.2**: Critical Triage Keywords ("Stroke") **shall** be validated via automated regression tests to ensure 100% detection.

- **SRD-15.3**: The system **shall** satisfy the Gherkin acceptance criteria defined in the PRD (Scenario: Stroke Patient Check-in).

16. Traceability Requirements

- **SRD-16.1**: Each requirement in this document **shall** map to a specific Feature or User Story in the **NeuroAssist MAX PRD v4.0**.
- **SRD-16.2**: Implementation of these requirements **shall** be verified in `jira_compliance_matrix.md`.