AI Scheduling Agent – Technical Approach (MVP)

Architecture Overview

A lightweight, deterministic agent implemented as a stateful Streamlit app. States: Greeting → Patient Lookup → Insurance Collection → Smart Scheduling → Calendar Booking → Confirmation & Exports → Intake Form Distribution → Reminders & Tracking. Data sources loaded at start; booking persists to Excel schedule.

Framework Choice

Chose plain Python + Streamlit for clarity and reliability for the MVP. LLM orchestration (LangGraph/LangChain) is left pluggable for future intent parsing but is not required to meet business logic and evaluation criteria.

Integration Strategy

Patient EMR simulated via CSV (50 synthetic records). Doctor calendar via Excel with 30-min slots; booking writes back to Excel. Email/SMS are simulated by writing logs to the exports folder; intake form PDF distributed post-confirmation.

Business Logic Highlights

New patient → 60 minutes (requires two consecutive 30-min slots). Returning patient → 30 minutes. Calendar filters by doctor, location, and date. Reminders: initial + 2 actionable reminders (forms status; visit confirmation/cancel reason).

Error Handling & Edge Cases

Input validation (name/DOB). Race-condition guard when booking taken slots. Empty-availability guidance. Safe fallbacks for missing contact data.

Deliverables Mapping

Demo UI covers full flow. Admin exports produced as Excel files. Code packaged with requirements and README. Technical approach captured in this PDF.