1. Problem Statement (Multimodal Clinical Insight Assistant)

Create a lightweight web app that empowers doctors to:

1. **Capture** a patient case summary by typing or dictation.
2. **Attach** lab reports (CSV/PDF) and radiology images (JPG/PNG/DICOM).
3. **Request** information via a simple spoken or typed command (e.g. “Show yesterday’s CBC for patient 47321” or “Read chest X-ray summary for case 45”).

In return, the system provides:

- A structured **SOAP note** (Subjective, Objective, Assessment, Plan)
- A list of **differential diagnoses**, recommended **investigations**, and **treatment options**
- **AI-driven interpretations** of each uploaded file
- A single **confidence score** for the combined output
- A concise **text reply** fulfilling any retrieval commands

---

2. Problem Statement (Multimodal Clinical Insight Assistant)

A. Backend

- **Case Management**
  - Persist case summaries and metadata.
  - Store uploaded files (local or cloud storage) and track their paths.
- **AI Orchestration**
  - Ingest new cases (text + optional lab + optional image), invoke AI models, and return a unified JSON result.
  - Process retrieval requests (text commands), parse intent/slots, fetch relevant data, and return both structured data and a brief text reply.

B. AI Agent Logic

- **Multimodal Fusion**
  - Convert lab files into structured data.
  - Generate captions or extract features from images.
  - Combine all inputs into one prompt to produce:
    - SOAP note
    - Differential diagnoses
    - Investigation & treatment suggestions
    - Per-file interpretations
    - Overall confidence score
- **Command Parsing**
  - Lightweight NLU (e.g. regex-based) to identify intents (“lab,” “image,” “case”) and parameters (ID, date).

C. Frontend

- **Interface Elements**
  - Text area for case entry (with optional speech-to-text toggle).
  - File-upload controls for labs and images.
  - Mic button or text field for retrieval commands.
- **Results Display**
  - Render the AI's JSON output in a clear, collapsible format.
  - Show the system's text reply alongside the data.
  - Simple feedback (thumbs up/down) to flag good vs. poor responses.

---

3. Easy Additional Features

1. **Basic Authentication**
   - Allow doctors to log in, so each user sees only their own cases.
2. **Audit Trail**
   - Track when cases are created, modified, and reviewed.
3. **Case Search & Filtering**
   - Let doctors search past cases by patient ID, date range, or keywords.
4. **Export to PDF**
   - Generate a downloadable PDF of the SOAP note and file interpretations.
5. **Feedback Dashboard**
   - Aggregate thumbs-up/down metrics to monitor overall AI performance over time.

---

4. Evaluation Criteria

Criterion	What to Look For
Backend & Data Layer	Clean design, secure file storage, proper data modeling
AI & Multimodal Reasoning	Reliable lab parsing, meaningful image interpretation, clear JSON schema
Command Handling	Accurate intent/slot extraction, correct data retrieval, concise replies
Frontend UX & Accessibility	Intuitive workflows, responsive design, keyboard & screen-reader support
Code Quality	Readable, well-organized code with basic type annotations
Basic Dev Setup	Simple start script or docker-compose for local run
Documentation	Clear README with setup/run instructions and sample dataset
Optional Features	Any additional enhancements implemented cleanly (authentication, search, export, etc.)

Why this matters: You're building more than an app—you're giving doctors a smarter, faster way to understand patient data. By melding text, labs, and images into one seamless tool, you'll speed decisions, ease workloads, and directly impact patient care. Dive in—your work here truly makes a difference.