AI-driven Medical Report Analysis

Introduction

The solution is an AI-assisted medical report analysis platform that converts uploaded laboratory reports, such as blood and urine reports, into structured biomarker data and an initial clinical summary for doctors to refer. The platform combines document parsing and OCR, deterministic biomarker/range logic, configurable clinical formulas, and an agentic reasoning layer.

The goal is not to replace clinicians or provide autonomous diagnoses. Instead, the application reduces the effort required to review repetitive laboratory data and provides physicians with a structured, evidence-based starting point for clinical assessment.

Furthermore, the platform automatically organizes extracted biomarkers by body-system, such as metabolic, cardiovascular, renal, hepatic, endocrine, hematologic, and inflammatory systems. This enables clinicians to quickly and comprehensively assess organ system health and multi-system physiological interactions

Client Details

Name: Confidential | **Industry:** Healthcare | **Location:** USA

Technologies

Layer	Technology	Role
Frontend	ReactJS	Report upload, processing status, results and doctor-facing UI
API / Backend	FastAPI	REST APIs, orchestration entry points and application services
PDF Processing	pypdf	Text extraction from machine-readable PDF reports
OCR	PaddleOCR	Text recognition for scanned/image-based reports
LLM	Claude Sonnet	Contextual reasoning and natural-language report generation
Agent Framework	LangChain	LLM/tool integration and reusable AI components

AI-driven Medical Report Analysis

Agent Orchestration	LangGraph	Stateful workflow, routing, validation and controlled execution
Knowledge / Rules	Application Database	Biomarkers, ranges, system combinations and formulas
Cloud	AWS EC2	Application runtime / backend deployment

Project Description:

Business Challenge

- Medical reports arrive in different PDFs and image layouts, making manual extraction inconsistent and time-consuming.
- The same biomarker may appear under different labels, units, or report formats.
- Doctors need to review many individual values together, especially when multiple biomarkers contribute to a system-level assessment.
- Rule-based formulas and combinations are difficult to apply consistently when done manually.
- The solution needed traceability from source report → extracted value → reference range → rule/formula → generated finding.

Solution Overview

The platform provides a workflow in which a user uploads one or more medical reports. The backend determines whether text can be extracted directly from the PDF or whether OCR is required. Extracted content is converted into normalized biomarkers with values and units. These biomarkers are evaluated against a database containing reference ranges, system-specific biomarker combinations, and configured formulas/rules. Claude Sonnet, orchestrated through LangChain and LangGraph, is used where contextual reasoning and report generation add value.

Example : Clinical Processing Scenario

Consider a blood report containing multiple cardiovascular-relevant biomarkers. The extraction layer identifies each biomarker and its measured value. The normalization layer maps report-specific labels and units to canonical definitions. The rule engine evaluates individual ranges and configured

AI-driven Medical Report Analysis

combinations. If sufficient evidence is available, the agent uses those structured findings to compose a concise initial summary highlighting relevant observations for the doctor.

The important design principle is that the LLM is not expected to invent numerical interpretation. The application supplies extracted values, reference information, and deterministic rule outputs as structured context.

Key Design Principles

- Separation of extraction, deterministic evaluation and language generation.
- Traceability of each finding back to extracted biomarker data and configured rules.
- Configurable knowledge base so biomarker ranges and system combinations can evolve independently.
- Human-in-the-loop review: the doctor remains responsible for clinical interpretation and final decisions.
- Graceful handling of heterogeneous report formats through PDF parsing plus OCR.
- Agent orchestration that can validate prerequisites and route tasks rather than relying on one unrestricted LLM call.

Security, Reliability and Clinical Safety Considerations

- Medical documents should be treated as sensitive data and protected with appropriate access controls, encryption, secure transport and retention policies.
- The generated report should clearly be presented as an AI-assisted draft rather than a definitive diagnosis.
- Numerical calculations and configured formulas should remain deterministic and testable outside the LLM.
- Extraction confidence and missing/ambiguous biomarkers should be surfaced rather than silently guessed.
- Audit logging should capture processing status, rule versions and report-generation events.
- Clinical validation and governance should be applied before production clinical use.

AI-driven Medical Report Analysis

Workflow

Step 1: Report Upload
The user uploads a medical report through the ReactJS web interface. The document is sent to the FastAPI backend.

Step 2: Text Extraction and OCR
For text-based PDFs, pypdf extracts embedded text. For scanned documents or images, PaddleOCR recognizes the report content.

Step 3: Biomarker Extraction
The extracted text is processed to identify biomarker names, measured values, units, and relevant context, with normalization to a canonical representation.

Step 4: Reference and System Evaluation
Normalized biomarkers are compared with configured reference ranges and system-specific biomarker combinations.

Step 5: Formula and Rule Evaluation
Configured formulas and clinical rules are evaluated using the relevant structured biomarker input

Step 6: AI-Assisted Report Generation
Claude Sonnet, coordinated through LangGraph, generates an initial readable report from the structured evidence for doctor review

AI-driven Medical Report Analysis

Architecture:

Figure 1 - High-level solution architecture

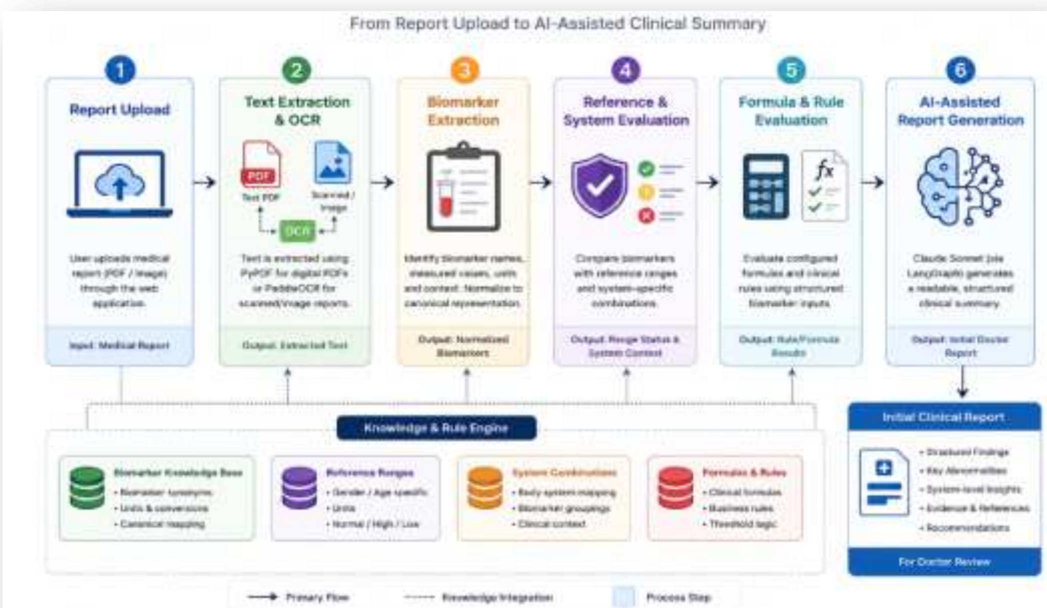


Figure 2 - End-to-end medical report processing workflow

AI-driven Medical Report Analysis

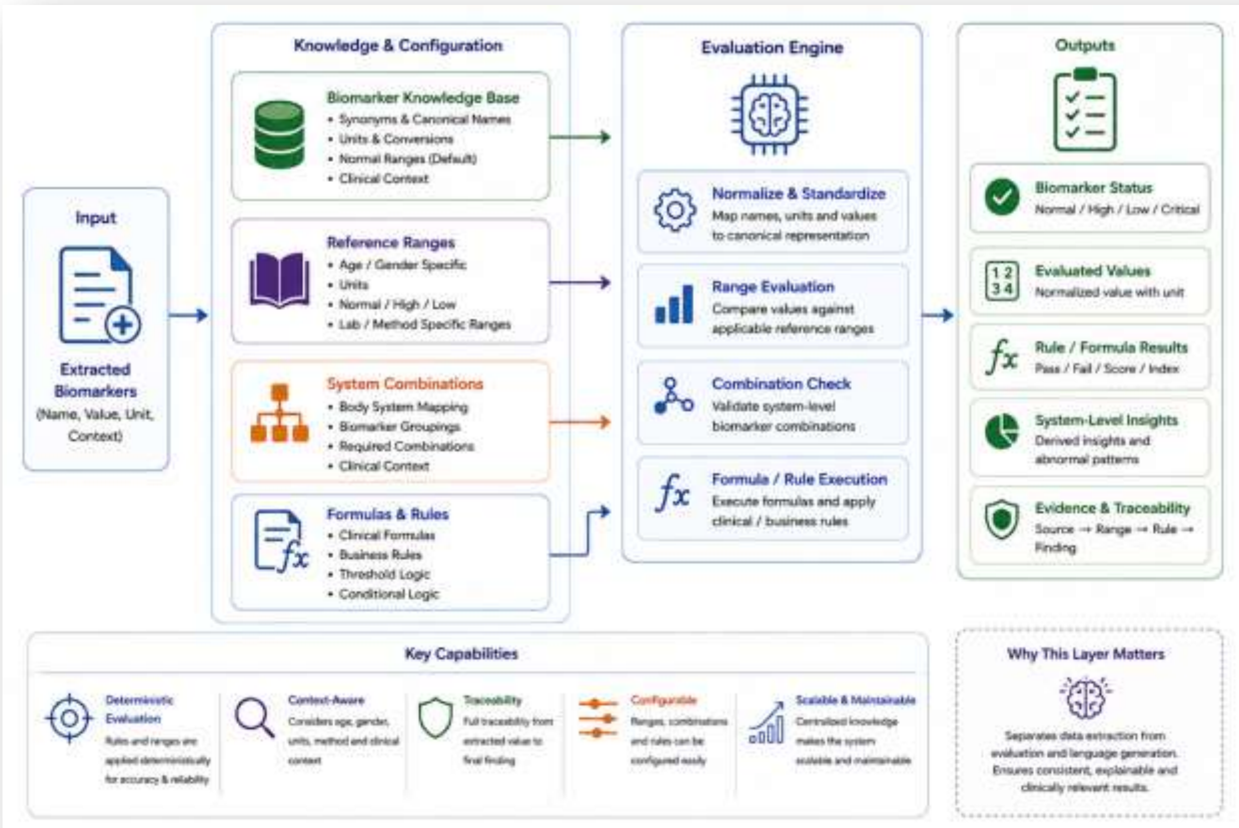


Figure 3 - Biomarker intelligence and rule evaluation

A key differentiator is that the platform does not rely only on an LLM to decide whether a value is abnormal. Structured knowledge is maintained for biomarkers, reference ranges, system-level combinations, and configured formulae. This creates a clear separation between data extraction, numerical evaluation, and natural-language generation.

AI-driven Medical Report Analysis

Agentic Architecture

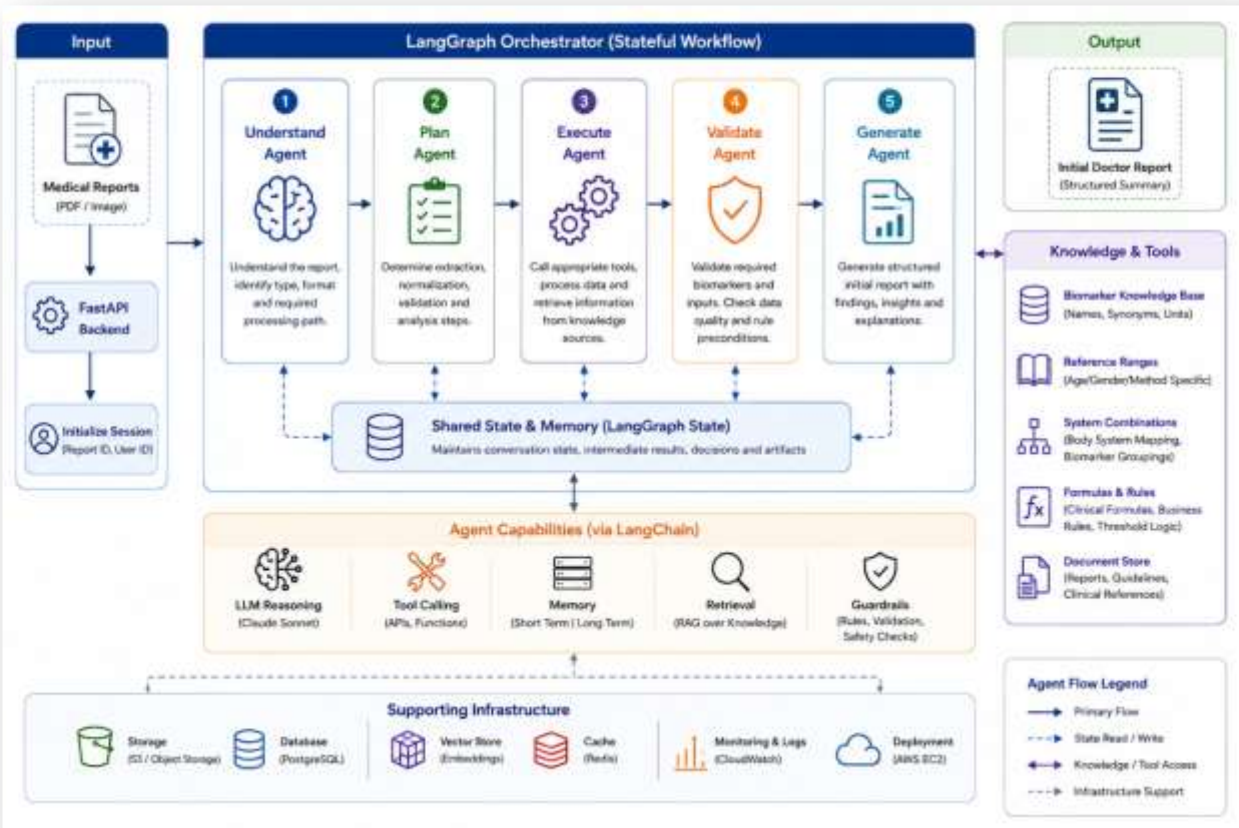


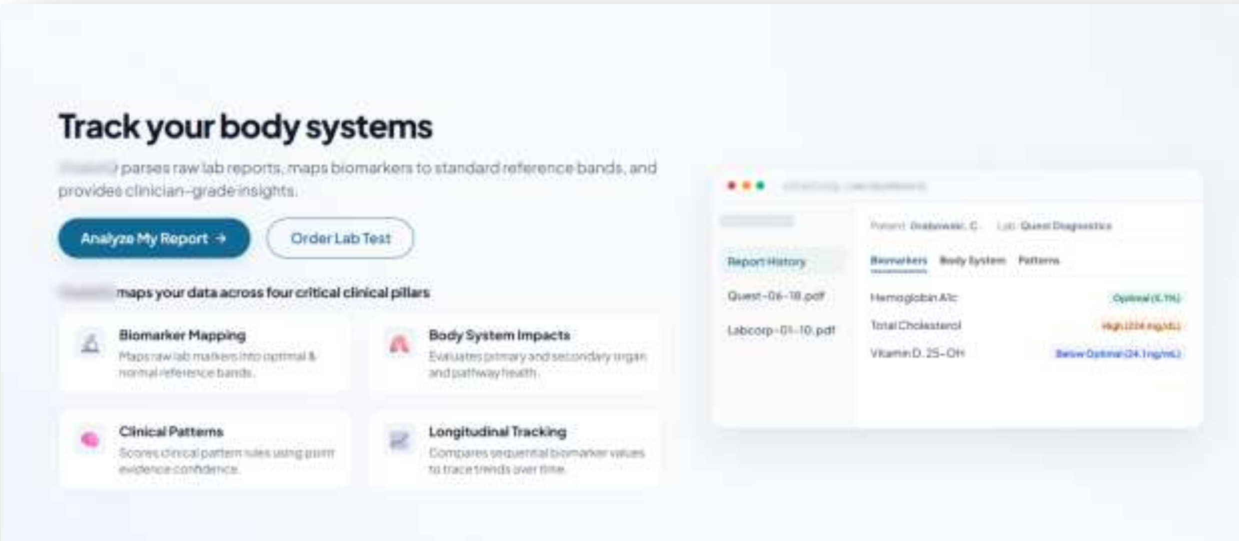
Figure 4 - LangGraph-based agent orchestration

LangGraph provides the orchestration layer for a controlled sequence of processing steps. LangChain provides reusable abstractions for LLM interaction and tool integration, while Claude Sonnet provides reasoning and language-generation capability.

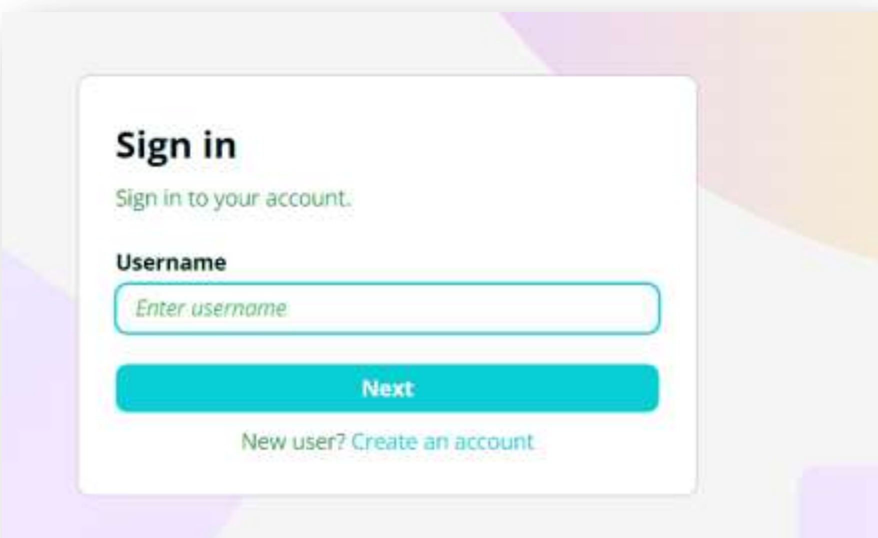
- Understand — identify the report and required processing path.
- Plan — determine extraction, normalization, validation and analysis steps.
- Execute — call the appropriate processing tools and knowledge sources.
- Validate — check that required biomarkers and inputs are available.
- Generate — create a structured initial report from validated results.

AI-driven Medical Report Analysis

Screenshots:

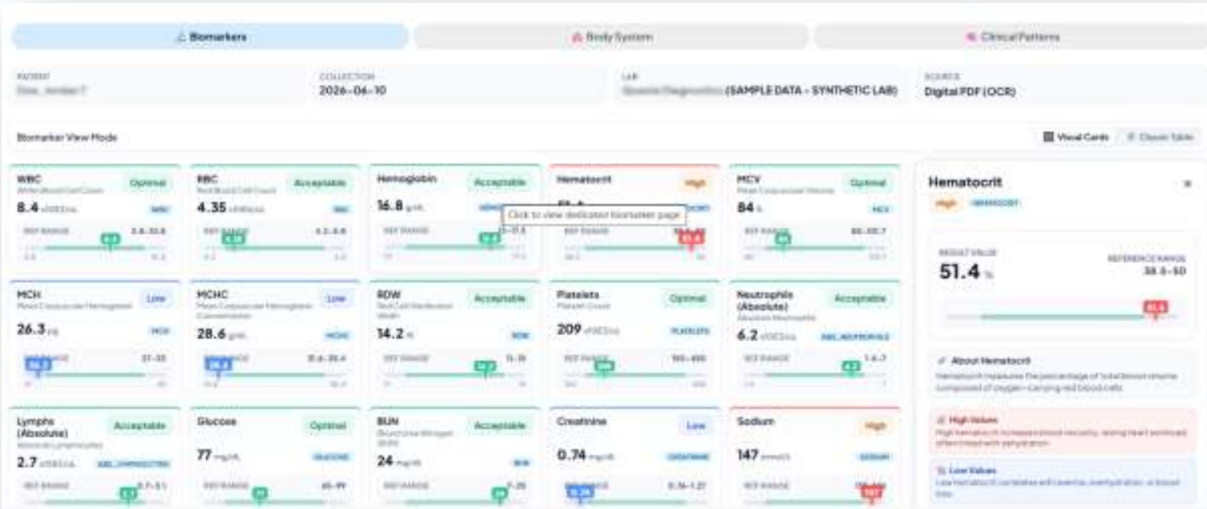


1: Landing Page

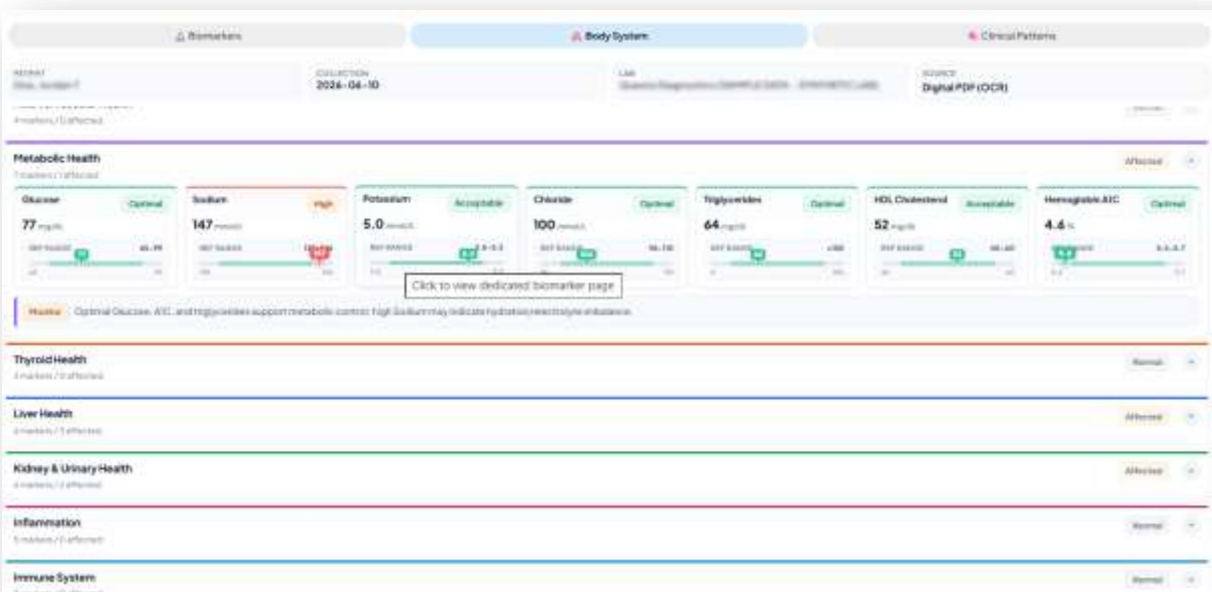


2: Login Page

AI-driven Medical Report Analysis



3: Biomarkers identified from lab report



4: Mapped Body System

AI-driven Medical Report Analysis

Business Impact:

- Reduces manual effort involved in reading and transcribing laboratory values.
- Creates a consistent structure for heterogeneous medical reports.
- Enables multi-biomarker and system-level analysis instead of isolated value review.
- Accelerates preparation of an initial doctor-facing summary.
- Provides an extensible foundation for adding biomarkers, systems, formulae, report types and analysis workflows.
- Creates a path towards longitudinal analysis as multiple reports become available

Future Enhancements:

- Patient-level longitudinal trend analysis across multiple reports and dates.
- Confidence scoring and source-location references for each extracted biomarker.
- Support for additional report types and document formats.
- Doctor feedback loops to improve extraction and report presentation.
- Role-based access control, enterprise audit trails and production observability.
- Containerized deployment and auto-scaling on AWS as workload grows.
- Versioned clinical rules and formula governance so that changes are auditable.

Conclusion :

The application combines document intelligence, structured biomarker knowledge, deterministic clinical logic and agentic AI into a practical medical-report analysis workflow. ReactJS provides the user experience, FastAPI exposes application services, pypdf and PaddleOCR handle heterogeneous document extraction, and Claude Sonnet with LangChain/LangGraph provide controlled reasoning and report generation. AWS EC2 provides a straightforward deployment foundation.

The resulting architecture helps doctors move from an unstructured laboratory report to a structured, reviewable initial report faster while keeping clinical responsibility, deterministic calculations and human review at the center of the workflow.