INFOSYS INTERNSHIP

PROJECT SHOWCASE 2026

INTELLIGENT ANALYTICS SYSTEM

MediQ

Intelligent Medical Report Processing

> Transforming complex, unstructured medical reports into clear, actionable insights through advanced OCR and Multi-Model AI analysis.

PV

Peruri Veera Venkata Durga Mahesh

Software Engineering Intern

94%
EXTRACTION ACCURACY

15+
REPORT FORMATS

~4.6s
AVG. PROCESSING TIME

SYSTEM ARCHITECTURE

Complete Processing Pipeline

MediQ Engine v1.0

Intelligent Routing

Automatically directs input to specialized parsers based on format detection.

3-Tier Fallback

Ensures reliability by switching extraction engines if primary fails.

Real-Time Telemetry

Live monitoring of processing latency across every stage of the pipeline.

TOTAL TIME

~4.6s

End-to-End

PROBLEM STATEMENT

From Complexity to Clarity

Comparison Analysis

The Challenge

Manual Process

Complex Medical Jargon

Difficult for patients to understand terminology.

Fragmented Formats

PDFs, images, and handwritten notes vary widely.

Zero Context

Raw data without interpretation or trends.

Time Consuming

Manual review takes significant professional time.

MediQ Solution

Intelligent Automation

Plain Language

Automated simplification of complex terms.

Universal Handling

One pipeline for all document types (OCR + AI).

AI Risk Analysis

Instant context, risk scoring, and next steps.

Near-Instant Results

Processing reduced to seconds per report.

15 Minutes → 30 Seconds

PROCESSING TIME REDUCED

Confusion → Clarity

USER EXPERIENCE TRANSFORMATION

TECHNICAL DEEP DIVE

Engineering the System

Modular Architecture v1.2

FRONTEND
User Interface
React · TypeScript · Tailwind CSS

BACKEND
API & Coordination
Python 3.9 · Flask · REST API

AI CORE
Logic & Reasoning
Primary: Gemini Pro → GPT-4o → Claude · *Auto-Fallback*

EXTRACTION
OCR & Parsing
Google Vision · Tesseract · EasyOCR

STORAGE
Persistence
SQLite · Session Store · Encryption

KEY DECISIONS

Type Safety
TypeScript prevents runtime errors and ensures contract adherence between heavy API data structures.

Modular Backend
Flask chosen over Django for lightweight integration with Python ML libraries and lower latency.

Resilient Strategy
Multi-model daisy chaining ensures uptime. If Gemini fails or timeouts, GPT-4o takes over instantly.

USER CENTRIC DESIGN

Designed for Real Healthcare Use

Dual Persona Strategy

Medical Professional

CLINICAL EFFICIENCY

High Volume Review
Processes dozens of complex reports daily without fatigue.

Quick Validation
Instant visibility of key metrics and abnormal trends.

Time Savings
Reduces review time from minutes to mere seconds.

Patient

CLARITY & CONFIDENCE

Non-Technical Access
Simple interface for users without medical training.

Actionable Clarity
Plain language explanations instead of raw, confusing data.

Health Trends
Visualizes progress over time to build health confidence.

Accessibility-focused design (WCAG 2.1)

Multilingual ready architecture

PROJECT POSITIONING

Beyond Typical Student Projects

Comparative Analysis

Typical Projects

COMMON SUBMISSIONS

CRUD-Focused
Basic database operations without complex logic or processing.

Single-Tech Stack
Limited to one language/framework, missing integration complexity.

No Resilience
Lacks error handling, telemetry, or fallback mechanisms.

THIS PROJECT

MediQ System

ENTERPRISE GRADE

End-to-End Pipeline
Full processing flow from input to actionable insight generation.

AI + Full-Stack
Seamless integration of frontend, backend, OCR, and Multi-Model AI.

Resilience & Telemetry
Production-ready with fallback logic and real-time monitoring.

Commercial Tools

MARKET SOLUTIONS

Expensive
High licensing costs make them inaccessible for smaller clinics.

Closed Ecosystem
"Black box" systems with limited customization options.

Vendor Lock-in
Difficult to migrate data or switch providers later.

> Built as a system, not just a submission.

PRODUCTION GRADE

Enterprise-Level Capabilities

System Features v2.1

Real-Time Telemetry

- Full observability pipeline tracking end-to-end latency.
- Monitors engine usage stats and per-stage bottlenecks.
- Logs cache hits vs. misses for optimization.

Metric: 50ms Granularity

Intelligent Caching

- Hash-based deduplication identifies identical reports.
- Prevents redundant processing of re-uploaded files.
- Short-term TTL for session persistence.

Impact: 90% Latency Drop on Re-runs

Multi-Model AI Fallback

- Automatic daisy-chaining: Gemini → GPT-4o → Claude.
- Triggers on API timeouts or malformed responses.
- Circuit breaker pattern prevents cascading failures.

Reliability: 99.9% Uptime

Confidence & Risk Scoring

- Assigns confidence score (0-100%) to every extraction.
- Auto-flags low-confidence data for human review.
- Color-coded risk indicators in final dashboard.

Safety: Human-in-the-loop Trigger

DEVELOPMENT JOURNEY

From Learning to System Ownership

6 Weeks Timeline

1 Benchmarking

Evaluated Tesseract vs. Google Vision. Optimized OCR preprocessing for medical fonts.

2 AI Reliability

Implemented multi-model fallback design. Handled hallucinations with confidence thresholds.

3 Frontend & UX

Refined state management for upload flows. Designed accessible, high-contrast dashboard.

4 Optimization

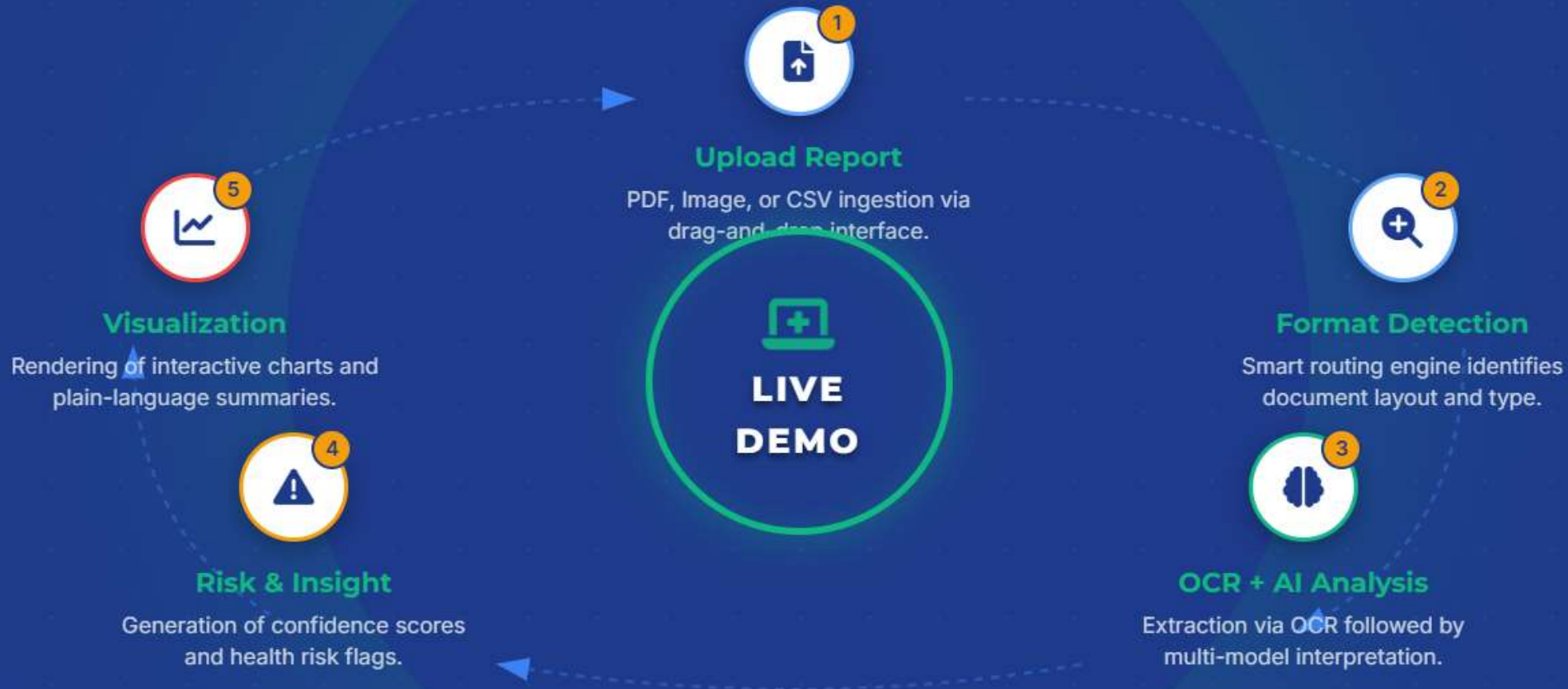
Telemetry integration for debugging. Reduced latency via caching and query optimization.

147 GIT COMMITS | **3,200+** LINES OF CODE | **6 Weeks** OF ITERATION

LIVE DEMONSTRATION

MediQ in Action

System Live Demo



Note: Demo is live. Screenshots and video backup available in case of network latency.

CLOSING NOTE

"I didn't build MediQ to impress a system. I built it to understand systems — and to make complexity clearer for real people."

THANK YOU FOR LISTENING

Questions & Discussion