
HealthIQ AI v5

Functional & Technical Design Document

---

1. Vision & USP

World's most advanced personalised biomarker analytics platform.

- **Mission**: Transform routine blood results into a deep, interconnected health intelligence journey — grounded in clinical science, enriched by AI, delivered through an elite UX.
- **Unique Selling Proposition (USP)**:
    - Every biomarker fully explained in biological and systemic context
    - Canonical-only downstream enforcement — no leaks of raw/non-canonical names
    - Network-based interpretation through **clusters** and **composite signatures**
    - Narrative, visual, and interactive insights — far beyond "PDF-like" reports
    - Clinically credible, evidence-based thresholds with confidence intervals

---

2. Functional Scope

User Workflow

1. **Data Ingestion**
    - Upload PDF, CSV, or manual entry of blood results
    - Complete structured lifestyle questionnaire
    - Metadata: age, sex, demographics
2. **Canonical Normalisation**
    - Map non-canonical names to canonical IDs via BiomarkerAliasResolver
    - Derived values auto-calculated (TG:HDL, HOMA-IR, etc.)
3. **Scoring & Cluster Analysis**
    - Each biomarker scored against evidence-based ranges
    - Cluster engines detect patterns (e.g. *Silent Inflammation Load*)
    - Composite signatures evaluated (e.g. *Longevity Resilience Index*)

4. **Insight Generation**
    - Insight orchestrator aggregates biomarker + cluster outputs
    - Gemini/Narrative AI produces personalised explanation
    - Recommendations surfaced with evidence and confidence levels
5. **Frontend Experience**
    - Real-time progress pipeline (Upload → Normalise → Score → Cluster → Insights)
    - Interactive biomarker cards, cluster radar, trend charts
    - Narrative cards with trust indicators, confidence, and lineage
    - Exportable reports and provider/family sharing

---

3. Technical Architecture

Backend (Python, FastAPI)

- **Data Models**: Pydantic immutable models (BiomarkerValue, BiomarkerPanel, UserContext)
- **Context Layer**: AnalysisContext with strict validation (frozen objects)
- **Canonical Enforcement**: _assert_canonical_only() at orchestrator boundary
- **Scoring Engine**:
    - Evidence-based thresholds (with ranges + confidence intervals)
    - Derived ratios and composite signatures
- **Insight Engine**:
    - BaseInsight contract → per-domain insights (Metabolic Age, Heart, Detox, Inflammation)
    - CompositeSignature layer for multi-biomarker USP patterns
- **Narrative AI**: Structured JSON → Gemini prompt → personalised summary
- **Database**: Supabase (Postgres) for user data, biomarker refs, signatures
- **APIs**: FastAPI with OpenAPI docs, <200ms target response

Frontend (Next.js + React + Tailwind + TypeScript)

- **Component Domains**:
    - Analysis/ → pipeline progress, real-time normalisation visual

    - Visualization/ → interactive panels, cluster radar, trends
    - Narrative/ → insight cards, recommendations, progress tracking
    - Collaboration/ → shareable reports, provider/family views
- **State Management**: React Query + Zustand
- **Mobile-First UX**: progressive disclosure, offline caching, accessibility config
- **Visualisation**:
    - react-gauge-chart for biomarker dials
    - D3.js/Chart.js for radar + trend graphs
    - Cross-correlation heatmaps

Cross-Cutting Services

- Notifications (email, in-app)
- Storage (Supabase buckets for uploads)
- Logging/Monitoring (Prometheus/Grafana)
- Security (JWT auth, RBAC for providers)

---

4. Sprint Roadmap

Sprint 0: Foundation Setup

- Repo setup (backend/, frontend/, docs/)
- CI/CD pipeline (tests + lint)
- Supabase/DB schema migration
- Dev tools: Cursor, Pre-commit hooks, mypy, pytest baseline

Sprint 1: Technical Foundation

- Implement canonical-only context architecture (as per Claude's doc)
- Replace legacy contexts with AnalysisContext
- 95%+ coverage on context pipeline
- Load + performance baseline established

Sprint 2: Clinical Depth

- Audit and update biomarker thresholds with evidence refs
- Add 15–20 advanced biomarkers (hormones, advanced lipids, micronutrients)
- Implement 3 new health domains (Hormonal Health, Micronutrients, Cardiovascular)
- Clinical advisory board recruited

Sprint 3: UX Excellence

- Build frontend progress pipeline, biomarker cards, cluster radar
- API standardisation with error handling
- Mobile optimisation and progressive loading
- Launch personalised recommendation engine (frontend)

Sprint 4: Intelligence Differentiation

- Implement composite signatures (Longevity, Fatigue, Silent Inflammation, Metabolic Instability, Detox)
- Build correlation engine + predictive analytics
- Deploy unique Health Intelligence Score
- Integrate AI Health Coach (Gemini-powered)

---

5. USP Integration: Why This Architecture Wins

- **Canonical Boundary Enforcement**: Prevents the technical debt problems of v4
- **Cluster & Signature Layer**: Encodes HealthIQ AI's unique systemic interpretation
- **Immutable Contexts**: Reliable, type-safe foundation (no silent state bugs)
- **Narrative Layer**: Gemini only interprets structured canonical outputs → no hallucinations
- **Elite UX**: Progress transparency + interactive exploration builds user trust and engagement
- **Clinical Credibility**: Evidence-backed thresholds + advisory board oversight

---

6. Success Criteria

- **Technical**:
    - 99.9% uptime
    - <200ms API response

    - 95% test coverage
- **Clinical**:
    - 75+ biomarkers
    - 12+ health domains
    - Advisory board + published validation
- **UX**:
    - 80% user return within 30 days
    - 4.5/5 satisfaction
    - 25% trial → paid conversion
- **Market**:
    - Top 3 awareness in health insights
    - 100k active users in 12 months

✅ This document is standalone — it links your **tech stack directly to your USP** (clusters + composite signatures as the moat). It tells devs **what to build (pipeline, canonical boundary, clusters, UX)**, **how to build (Pydantic models, Next.js state, Supabase)**, and **when to build (Sprint roadmap)**.

Would you like me to now **package this into a polished PDF functional + technical design spec** so you can circulate it with the team as an official blueprint?