Functional Specification Document: CuraData – AI-Powered Preventive Health App

Product Overview

CuraData is a digital health app designed to decode users' lab results from blood, urine, saliva, and stool tests using AI. It combines natural language processing (NLP), computer vision (CV), and cultural awareness to offer personalized health insights. The platform focuses on accessibility, especially for underserved communities, and bridges modern lab data with ancestral healing perspectives.

Core Functionalities

User Authentication & Profiles

- Sign up/log in with email, password, or Google
- User role: patient (default), practitioner (future phase)
- Store user preferences (language, culture, symptom focus)

File Upload & Lab Input

- Upload lab reports (PDF, JPG, PNG, CSV)
- Manual input option for specific biomarkers (e.g., Vitamin D, Iron, TSH)
- File types routed to parsing layer (OCR or structured CSV handler)

AI Analysis Layer

- NLP + CV model processes lab files
- Extract markers from CMP, CBC, Vitamin D, Iron, Thyroid, Microbiome
- Provide optimal (not just standard) range interpretations
- Match findings with herbal, lifestyle, and nutritional suggestions

Personalized Health Report

- Summarized insights (text + visual)
- “What it means” + “What to do next”

- Available in English, Spanish, and Spanglish
- Stored per user in Firebase database

Wellness Tracker (Phase 2)

- Track symptoms, changes, and responses over time
- Code Basic Health Protocol
- Tag insights to specific uploads
- Optional chatbot integration

Data Architecture (Firebase)

Firebase Collections (confidential)

- /users: User profile data
- /lab_reports: Metadata + file paths
- /recommendations: AI-generated insights linked to lab reports
- /uploads: Raw files per user

Cloud Storage

- /uploads/{userId}/{filename}: PDF, image, CSV files

Security Rules

- Only authenticated users can access their own data
- Files and reports cannot be shared without permission
- Admin access limited to flagged or research accounts

AI Models & APIs

- GPT (via OpenAI or local LLM) for language generation
- OCR (e.g., Tesseract or Google Vision API) for scanned file parsing
- Custom health prompt templates based on biomarker categories

- Optional: API for database of functional medicine ranges

Frontend UI (MVP)

- Upload page
- Lab history dashboard
- Insights & recommendations panel
- Language toggle (EN/ES)
- Simple mobile-responsive interface (React or Flutter)

Milestones & Phases

Phase	Goals
Phase 1	Define scope, set up Firebase, gather sample labs, train/test basic prompts
Phase 2	Add frontend interface, enhance NLP model, create exportable report
Phase 3	Add wellness tracker, multilingual chatbot, user feedback loop

Testing Plan

- Unit test upload, parsing, and database write functions
- Validate GPT output accuracy with 10 sample labs
- Manual QA for UI upload flow and interpretation feedback

Deployment & Access

- Firebase backend
- GitHub repo public with open-source license
- Future: Deploy to web via Firebase Hosting or Vercel