Miguel Ángel Pascual Morante

Robotics Engineer and developer of AI and Computer Vision solutions

Bachelor's Thesis · Robotics Engineering, University of Alicante (2026)

linkedin.com/in/miguelangelpascualmorante | bioxs.app | mpascualmorante@gmail.com

A mobile app that brings a person's health data together in one place: activity and wearables, nutrition, habits and medical reports and, with artificial intelligence, turns it into clear analysis and personalised recommendations.

The project

BIOXS is a personal mobile health application developed as a Bachelor's Thesis. Its goal is to help a person **understand their own health**: it brings together what is usually scattered across a smartwatch or fitness band, nutrition apps, reports from tests such as laboratory work, and notes about medication or symptoms, and turns it into understandable readings and personalised recommendations.

BIOXS goes beyond displaying charts. A **deterministic analytics engine** processes the data and calculates trends, anomalies and relationships without delegating the figures to a generative model. The architecture also includes a personal **Knowledge Graph** that retains findings together with their time window, evidence, confidence and provenance. On top of that calculated context, an **artificial intelligence assistant** provides natural- language explanations, answers questions and uses the medical documentation supplied by the person.

The challenge

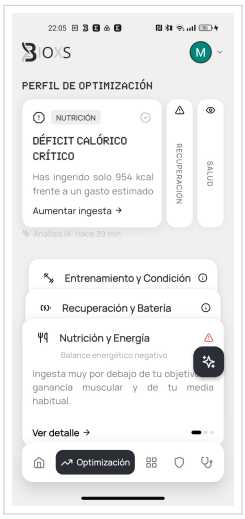
Personal health data is **fragmented and rarely actionable**: steps and sleep live in one app, lab results in a PDF, meals in another, and nothing is interpreted together. The challenge for BIOXS was twofold: to **unify** very different sources into a single experience, and to get the AI to help people better understand their health and make informed decisions, while protecting the privacy and security of their data.

BIOXS is an informational tool: it does not replace a healthcare professional, and the associations it detects do not demonstrate causality.

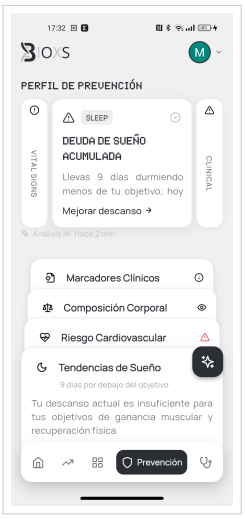
Key features

The app is organised into **five areas** (Home, Optimisation, Prevention, Chronic and Actions) covering everything from the daily summary to long-term chronic-condition tracking:

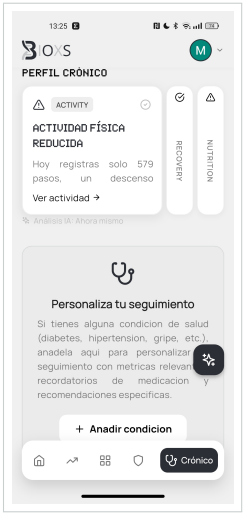
Home



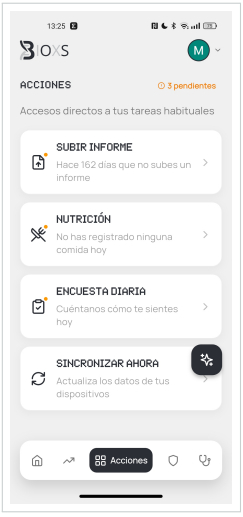
Optimisation



Prevention



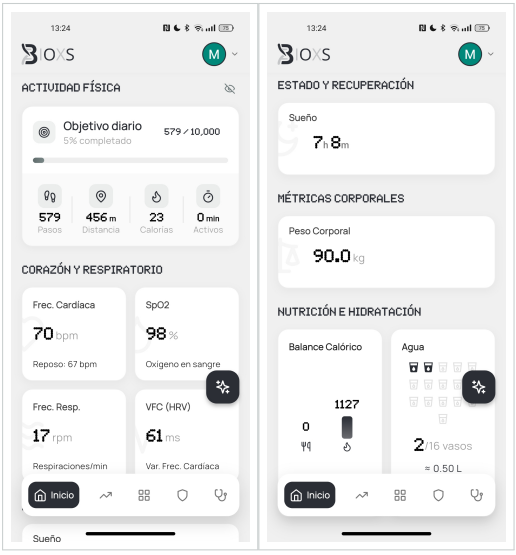
Chronic



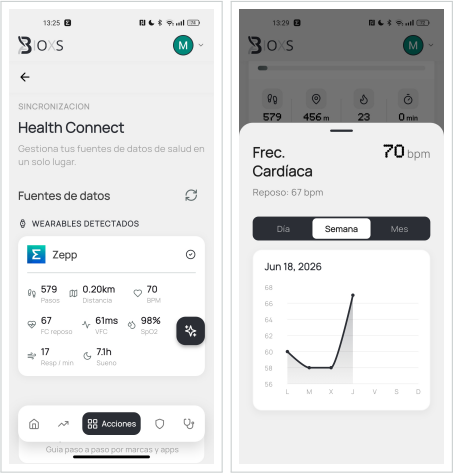
Actions

On top of that structure sit the key features:

- **Daily health dashboard.** The home screen distils the person's status into a global *score* and metric cards (activity, sleep, heart rate, nutrition, hydration…). Each card opens the indicator's full history and its evolution over time, so a single glance is enough to see how those metrics are doing today and how they have evolved over time.



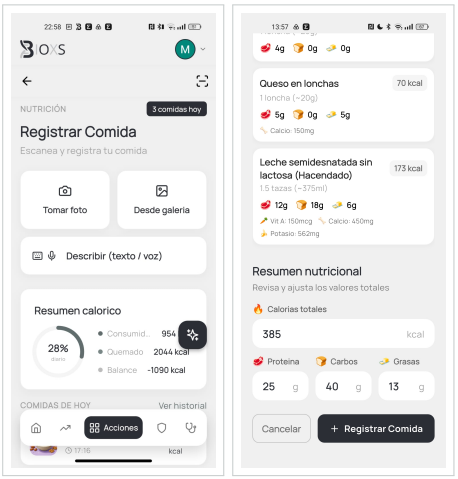
Daily health dashboard



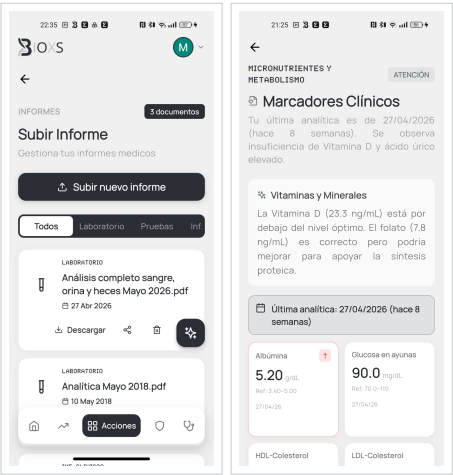
Sync and metrics

- **Automatic wearable tracking.** Through **Health Connect**, the app syncs steps, sleep, heart rate, calories, workouts and other signals from the user's watch or band, both in active use and in the background. Data is aggregated per day and kept so that medium- and long-term trends can be analysed.

- **AI nutrition logging.** Meals can be logged by voice, text or **taking a photo**: the AI recognises the dish and estimates its calories, macro- and micronutrients, which the user reviews and corrects before saving. It also generates recipes that match the user's goal and preferences using the ingredients available in their fridge or pantry, plus a base of *personalised products* (the ones the user usually buys) to make the calculations more accurate.



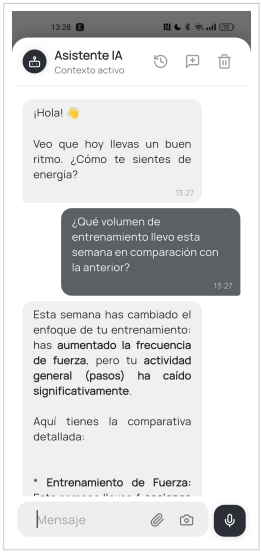
Logging and nutrition summary



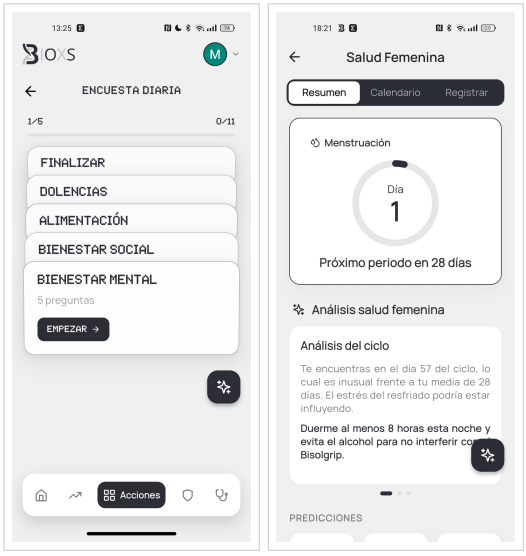
Upload and clinical markers

- **Smart medical reports.** The user uploads compatible clinical documents (lab results, test reports, clinical history, discharge or consultation reports) as PDF or photo. The app **sanitises** them to protect personal data, uses AI to extract the relevant information (biomarkers, mentioned diagnoses and guidance) and splits it into chunks indexed with *embeddings*, so they can be retrieved by similarity through **RAG** (*Retrieval-Augmented Generation*). This way, both the health analysis and the chat rely on the **clinical documentation supplied** by the person rather than on generic knowledge.

- **Contextual conversational assistant.** An AI chat with access to the user's medical profile, metrics and reports, that keeps the conversation context and accepts attachments. What sets it apart from a generic chatbot is ***tool-calling***: instead of answering only from what it «remembers», the model can *invoke tools* that query the person's real data in real time (meals, daily metrics, lab results, workouts, reports or the physical and mood state captured in the surveys) and retrieve the relevant report fragments through **RAG**, building the answer from them.



AI assistant



Daily survey and women's health

- **Women's health and habits.** Menstrual-cycle and symptom logging, integrated with the rest of the metrics, plus a daily physical-and-mood survey and meal, medication and report reminders delivered as notifications that take the user straight to the right action inside the app.

- **Weekly and monthly progress reports.** The app lets users browse, preview and download closed-period PDF reports. They compare each period with the previous one and combine metrics, nutrition, training and laboratory data; when graph relationships are supported and valid at the cut-off date, they can also be included.

- **Manual capture and detailed workouts.** Alongside Health Connect, users can record weight, glucose, blood pressure, hydration and sports sessions with activity, duration, intensity, distance, energy and notes; histories and charts combine manual and synced data.

- **Profile, security and data control.** Goals, units, sync and notifications are managed alongside biometrics, per-user isolation, document sanitisation and account deletion.

Application architecture

The three diagrams present the BIOXS architecture from complementary perspectives: the components and their communications, the path that turns data into traceable results, and the processing of medical reports so they can be incorporated into the analysis and AI context.

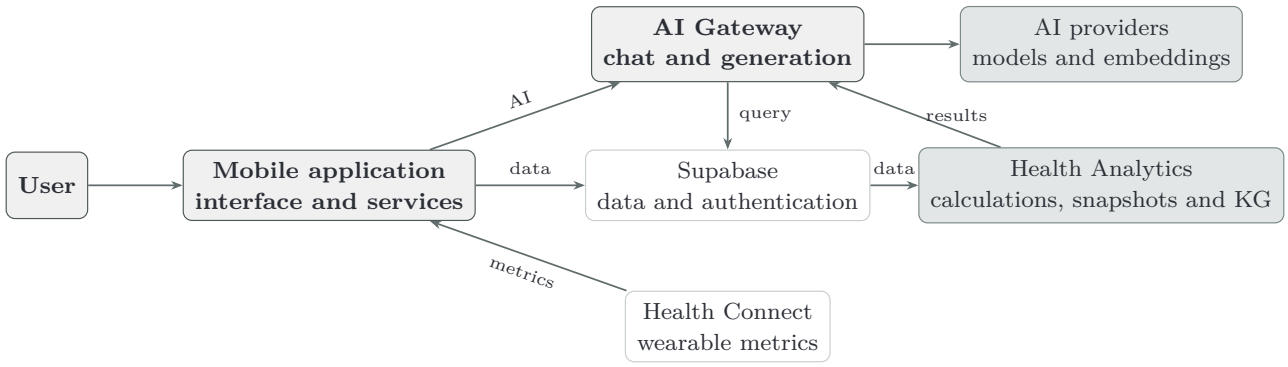


Diagram 1 · System components and communication. This diagram shows the main components of BIOXS and how they communicate. The user interacts with the Flutter app; Health Connect sends it wearable metrics, while Supabase authenticates the user and stores their data. Health Analytics reads that information to perform calculations and maintain the snapshots and Knowledge Graph. When an explanation or conversational response is needed, the AI Gateway connects the application to the artificial intelligence providers.

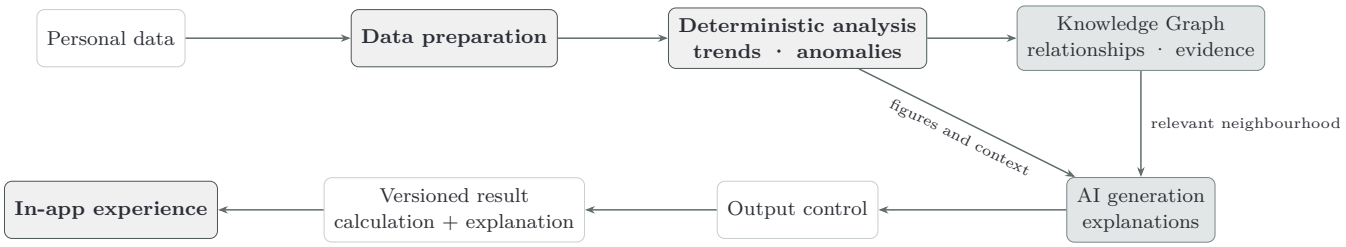


Diagram 2 · From data to a traceable recommendation. This flow represents how data becomes a recommendation. Health Analytics checks the quality of the information and creates a deterministic snapshot with trends, anomalies, correlations and scores. It then updates the relationships that have sufficient evidence in the Knowledge Graph. The AI Gateway explains those results, the output is validated and saved with its version before Flutter presents it as scores, recommendations and reports.

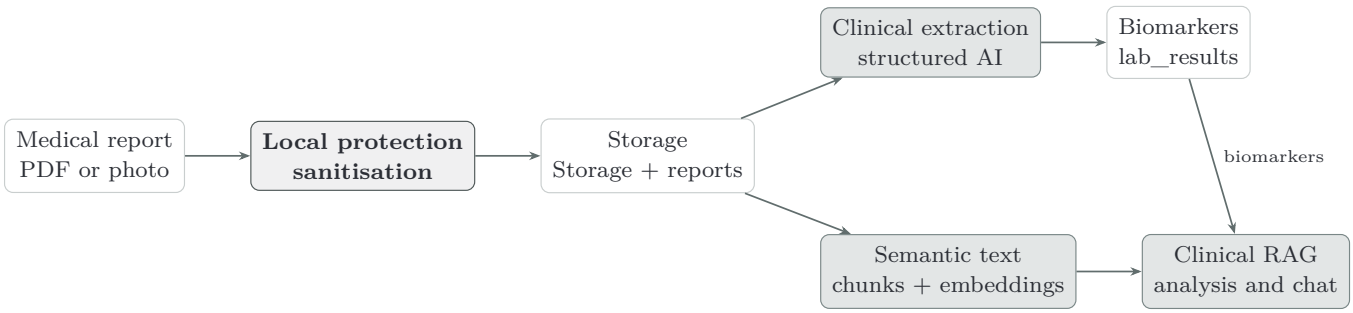
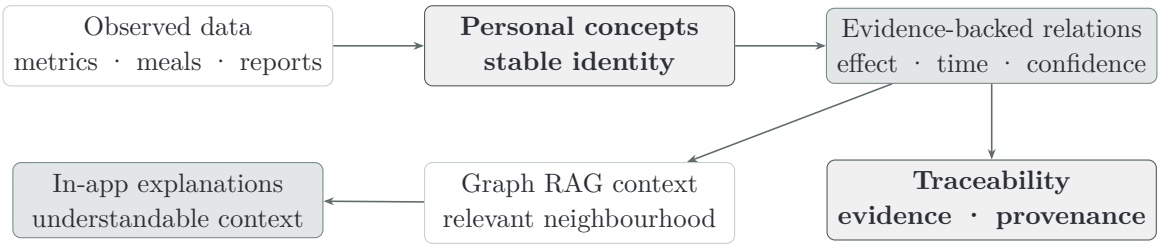


Diagram 3 · From the medical report to AI context. This diagram explains what happens when a person uploads a medical report. First, the file is sanitised and stored securely. Structured data such as biomarkers is then extracted on one side, while text fragments with *embeddings* are produced on the other. RAG retrieves only the fragments related to each query and combines them with the biomarkers to provide context for the health analysis and chat.

Knowledge Graph: longitudinal health memory

The **Knowledge Graph** turns a fragmented history into a personal network that connects concepts such as sleep, heart-rate variability (HRV), stress, activity, nutrition, laboratory results, symptoms, conditions and medication. It does not replace the original data: Supabase remains the source of truth, while the graph represents only stable concepts, personal entities and relationships supported by calculated evidence.



Each relationship answers four questions: *which* signals are connected, *when* the pattern was observed, *how* it was calculated and *how robust* it is. It therefore retains the method, time window, sample size, effect, lag, confidence, status and provenance. New runs can reinforce, contradict, replace or expire a relationship, allowing the knowledge to evolve without losing traceability.

Graph RAG retrieves only the relevant neighbourhood of the graph to contextualise the chat, reports and explanations. The LLM can communicate those findings, but it cannot calculate effects, promote relationships or turn an observational association into causality. The deterministic catalogue covers more than 55 signals and more than 45 predefined pairs; the discovery of new relationships requires multiple-testing correction, *bootstrap*, temporal stability and independent replications before they are incorporated as supported knowledge.

Key technologies

Mobile app

- **Flutter**
- **Dart**
- **Health Connect**

Data & backend

- **Supabase**
- **PostgreSQL**
- **Node.js / TypeScript**

Python · FastAPI

Artificial intelligence

- **Vertex AI**
- **OpenRouter**

Implemented techniques

RAG
Tool-calling
Embeddings
Knowledge Graph / Graph RAG
Guardrails

Web frontend & analytics

- **Next.js**
- **React**
- **Tailwind CSS**
- **PostHog**

Observability

- Sentry
- Better Stack
- PostHog

AI-assisted development

- Claude Code
- Codex
- Antigravity

AI agents used as support during development

What this project demonstrates

BIOXS is a complete product (app, backend, database, AI and distribution) designed and built end to end by a single person. The work demonstrates a set of technical and product skills directly applicable on a team:

- **Traceable AI and knowledge:** multimodal models with RAG, Graph RAG, *tool-calling*, an evidence-backed Knowledge Graph and guardrails, while calculations and evidence levels remain outside the LLM's control.
- **Clean, maintainable architecture:** design of the system in layers (UI, state, services and backend) with clear responsibilities and a predictable data flow, so each part can be understood, tested and evolved on its own.
- **Real mobile development:** development of a functional Flutter app with a production architecture, navigation, state management, background sync, wearable integration, camera, voice and notifications, including the handling of permissions, long-running processes and intermediate states.
- **Backend and data, end to end:** a secure relational model, Node/TypeScript gateway and reproducible Python engine that persists snapshots, reports, relationships and evidence, authenticating every request and isolating each user's data.
- **Security & privacy by design:** responsible handling of sensitive health data from the start: per-user isolation with *Row Level Security*, PKCE authentication, document sanitisation and data minimisation.
- **Product mindset and autonomy:** coverage of the whole cycle, from code development and AI integration to user experience, backend, observability, public landing and distribution.