



Hygieia

Health and Fitness Telemetry Mobile App

Requirements Specification

by: Matheus Arenivas

March 27, 2025

Contents

1	Introduction	2
2	Overall Description	2
2.1	Product Perspective	2
2.2	Product Functions	2
3	System Features	2
4	Prototyping	3
5	Tech Stack	5
5.1	Frontend	5
5.2	Backend	5
5.3	Machine Learning + AI	5
5.4	APIs and Integrations	5
5.5	Deployment + DevOps	5
5.6	Monitoring and Analytics	5
6	Milestones	6

1 Introduction

Hygeia is a mobile health and fitness telemetry application designed to:

- Collect and visualize sleep, heart rate, activity, and dietary data from wearable health devices (e.g., Oura Ring, Apple Watch).
- Use AI/ML to analyze user behavior and provide smart health insights such as sleep predictions, optimal workout times, and recovery recommendations.
- Allow users to log meals, track nutrition, and receive diet recommendations.
- Offer personalized fitness plans with AI-powered coaching and progress tracking.
- Function as a digital health assistant improving wellness, rest, and physical performance.

2 Overall Description

2.1 Product Perspective

Hygeia serves as a centralized health dashboard integrating data from multiple third-party health APIs. The system will be cross-platform and mobile-first using React Native.

2.2 Product Functions

- Aggregate data from Oura, Apple HealthKit, and Google Fit.
- Provide visualizations for sleep, heart rate, and activity.
- Nutrition logging and calorie tracking.
- AI-driven fitness and meal plan recommendations.
- Machine learning predictions for sleep patterns.

3 System Features

- **Telemetry Sync:** Automatically fetches health data.
- **User Dashboard:** Displays real-time health metrics.
- **Meal Tracker:** Enables food logging and macro analysis.
- **AI Assistant:** Recommends workouts and meal plans.
- **Predictive Engine:** Uses ML to forecast sleep and recovery.

4 Prototyping

- **Class Diagram:**

View Class Diagram on LucidChart

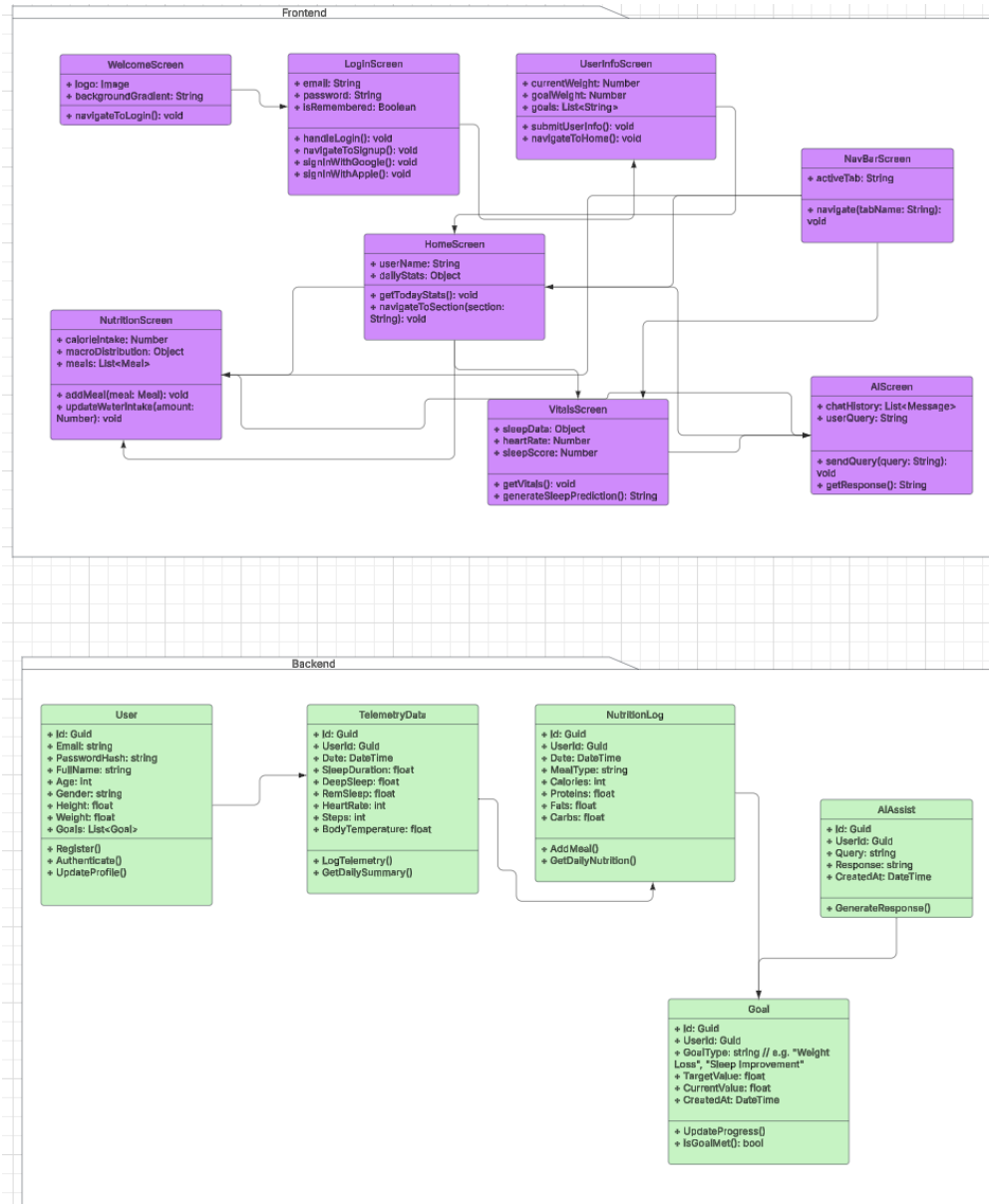


Figure 1: Class Diagram of Hygeia System

- **UI/UX Design:**
View UI/UX Prototype on Figma

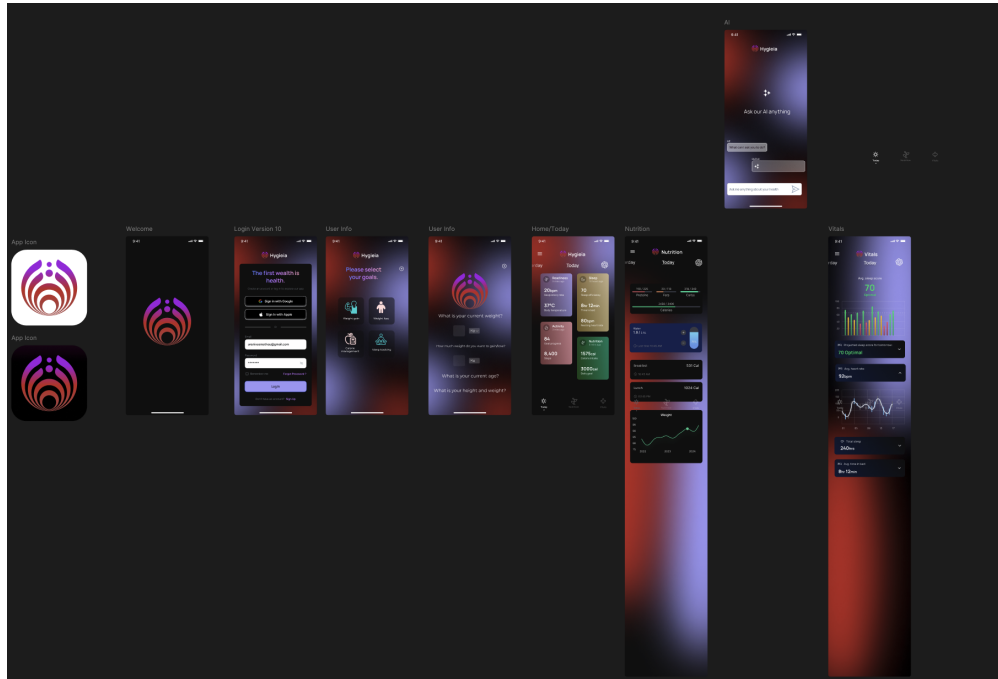


Figure 2: UI/UX Design Prototype

5 Tech Stack

5.1 Frontend

- React Native (w/ Expo)
- TypeScript
- NativeWind (Tailwind CSS for React Native)
- Axios or Fetch API

5.2 Backend

- ASP.NET Core Web API
- C# with Entity Framework Core
- PostgreSQL or MySQL

5.3 Machine Learning + AI

- Python (scikit-learn / PyTorch / TensorFlow)
- FastAPI / Flask for serving predictions

5.4 APIs and Integrations

- Oura API
- Apple HealthKit / Google Fit
- Nutritionix / Spoonacular
- OpenAI API

5.5 Deployment + DevOps

- Docker
- Render / Railway / Azure / AWS
- Expo Go or EAS Build

5.6 Monitoring and Analytics

- Sentry (error tracking)
- Firebase / Supabase (optional auth/analytics)

6 Milestones

- **Phase 1: Core Integration and Mobile UI**
 - Set up React Native frontend using Expo.
 - Integrate with the Oura API to fetch sleep and heart rate data.
 - Display data in a clean, real-time dashboard.
 - Ensure seamless communication with the ASP.NET backend.
- **Phase 2: User Authentication and Data Persistence**
 - Implement secure login and registration (OAuth or email/password).
 - Use PostgreSQL (or similar) to store user data and telemetry history.
 - Structure the backend to support multi-user tracking.
- **Phase 3: Nutrition Tracking Module**
 - Allow users to log meals and track macronutrients.
 - Integrate Nutritionix or Spoonacular API for food search and analysis.
 - Display nutrition summaries and trends in the app.
- **Phase 4: Sleep Prediction Engine (Machine Learning)**
 - Train an ML model using historical sleep data.
 - Predict future sleep quality, REM/deep/light sleep ratios.
 - Deploy the model with a lightweight Python API (FastAPI or Flask).
- **Phase 5: AI-Powered Fitness and Diet Planning Assistant**
 - Build an AI assistant that recommends meals and workouts.
 - Tailor suggestions to user goals, data, and performance trends.
 - Optionally integrate OpenAI for natural language responses.