MONITORAGGIOSALUTE.COM

PIATTAFORMA DI TELEMONITORAGGIO SANITARIO

I parametri vitali sotto controllo, in tempo reale.

Un sistema cloud che misura, raccoglie e sorveglia i valori basali di pazienti e anziani fragili — con allerta automatica quando qualcosa esce dalla norma.

Web + Mobile
DUE APPLICAZIONI FULL-STACK

Cloud-native
NEXT.JS · SUPABASE · VERCEL

01 — IL PROBLEMA

Il monitoraggio della salute fragile è frammentato e arriva tardi.

Episodico

I valori si rilevano solo in ambulatorio, una tantum. Tra una visita e l'altra non c'è alcuna fotografia continua del paziente.

Cartaceo

Misure annotate a mano, sparse tra dispositivi diversi. Dati difficili da aggregare, condividere e confrontare nel tempo.

Reattivo

Si interviene quando il problema è già conclamato. Il segnale precoce — una saturazione che scende, una febbre che sale — si perde.

La domanda | Come dare a famiglie, RSA e operatori una **vista continua e affidabile** dei parametri vitali, che **avvisi da sola** quando serve intervenire?

02 — LA SOLUZIONE

Un ciclo chiuso: dal dito del paziente all’allerta sullo smartphone del caregiver.

01

Misura

Dispositivi medici Bluetooth rilevano i valori vitali in pochi secondi, a casa o in struttura.

→

02

Carica

Ogni misura viene salvata nel cloud e storicizzata, associata al paziente e al dispositivo.

→

03

Valuta

Un motore di regole confronta ogni valore con le soglie cliniche personalizzate.

→

04

Allerta

Se il valore è fuori norma parte la notifica a staff e familiari, in automatico.

Nessun passaggio manuale tra la misura e l'allarme: il sistema lavora da solo, 24 ore su 24.

03 — IN SINTESI

Tre capacità, una piattaforma.

Monitorare

Parametri vitali, storico continuo, grafici per paziente e cruscotto multi-paziente per il monitoraggio di più utenti

Health Monitor · Statistiche

Allertare

Motore di regole con soglie warning/critiche, anti-duplicati ed escalation. Notifiche email, sms, e chiamate pre-registrate automatiche.

Alert Engine · Dispatcher

Proteggere

App mobile con GPS e geofencing per anziani (localizzazione e soglia di mobilità), videochiamata integrata per consulto diretto.

App Flutter · Jitsi · BLE

04 — ARCHITETTURA

Quattro livelli, un flusso unico di dati.

LIVELLO 1 · DISPOSITIVI

Dispositivo indossabile (BLE)

Promemoria farmaci sullo smartphone

Piano terapeutico impostato da remoto dal caregiver

Alert pochi minuti prima dell'assunzione; il paziente spegne l'allarme a farmaco preso e l'esito è registrato in cloud

Localizzazione del paziente via GPS

Zona sicura circolare intorno alla posizione, con alert se il paziente ne esce

Bluetooth LE

→

LIVELLO 2 · ACQUISIZIONE

App Web (PWA)
Web Bluetooth · kiosk RSA

App Mobile
Flutter · BLE · GPS

HTTPS / REST

→

LIVELLO 3 · CLOUD

API Next.js
37+ endpoint serverless

PostgreSQL
Supabase · storico misure

Alert Engine
valutazione + dispatcher

→

LIVELLO 4 · SORVEGLIANZA

Controllo multiutente

Notifiche email & SMS

Telefonate preregistrate

Caregiver & famiglia

escalation

05 — PARAMETRI VITALI

Cinque segnali, valutati a ogni misura.

Frequenza cardiaca

bpm

heart_rate

Saturazione ossigeno

SpO_2 %

spo2

Posizione GPS

Geolocalizzazione

Farmacologia

Alert per il paziente

Video chiamata

Chat video

Soglia warning · Soglia critica — ogni parametro è valutato verso il basso e verso l'alto.

06 — CONNETTIVITÀ DISPOSITIVI

Parlare con l’hardware, dal browser e dall’app.

Web Bluetooth

Lo scanner si connette ai dispositivi medici direttamente dal browser — nessuna app da installare per misurare in struttura.

BLE nativo (Flutter)

Su mobile la comunicazione passa per `flutter_blue_plus`, con scansione, connessione e gestione delle notifiche di caratteristica.

Protocollo del dispositivo decodificato

Comandi e UUID del dispositivo documentati per reverse engineering: HR e SpO_2 in tempo reale, storico, sonno, HRV.

```
// wearable_service.dart
SERVICE_UUID = "6e40fff0-…-e50e24dcca9e"
CHAR_WRITE = "6e400002-…"
CHAR_NOTIFY = "6e400003-…"

// comandi BLE
CMD_GET_REALTIME_HR = 0x69
CMD_GET_REALTIME_SPO2 = 0x6A
CMD_GET_HRV = 0x39
CMD_GET_STEPS = 0x43

// dispositivi rilevati via scan BLE
```

07 — FLUSSO UTENTE · PAZIENTE

Semplice come misurarsi la febbre.

1 **Accesso con codice fiscale**

Login essenziale, senza username complessi. Validazione del codice fiscale lato sistema.

2 **Misurazione guidata e richiesta da remoto**

L'utente avvia il dispositivo; i valori arrivano via Bluetooth e compaiono a schermo.

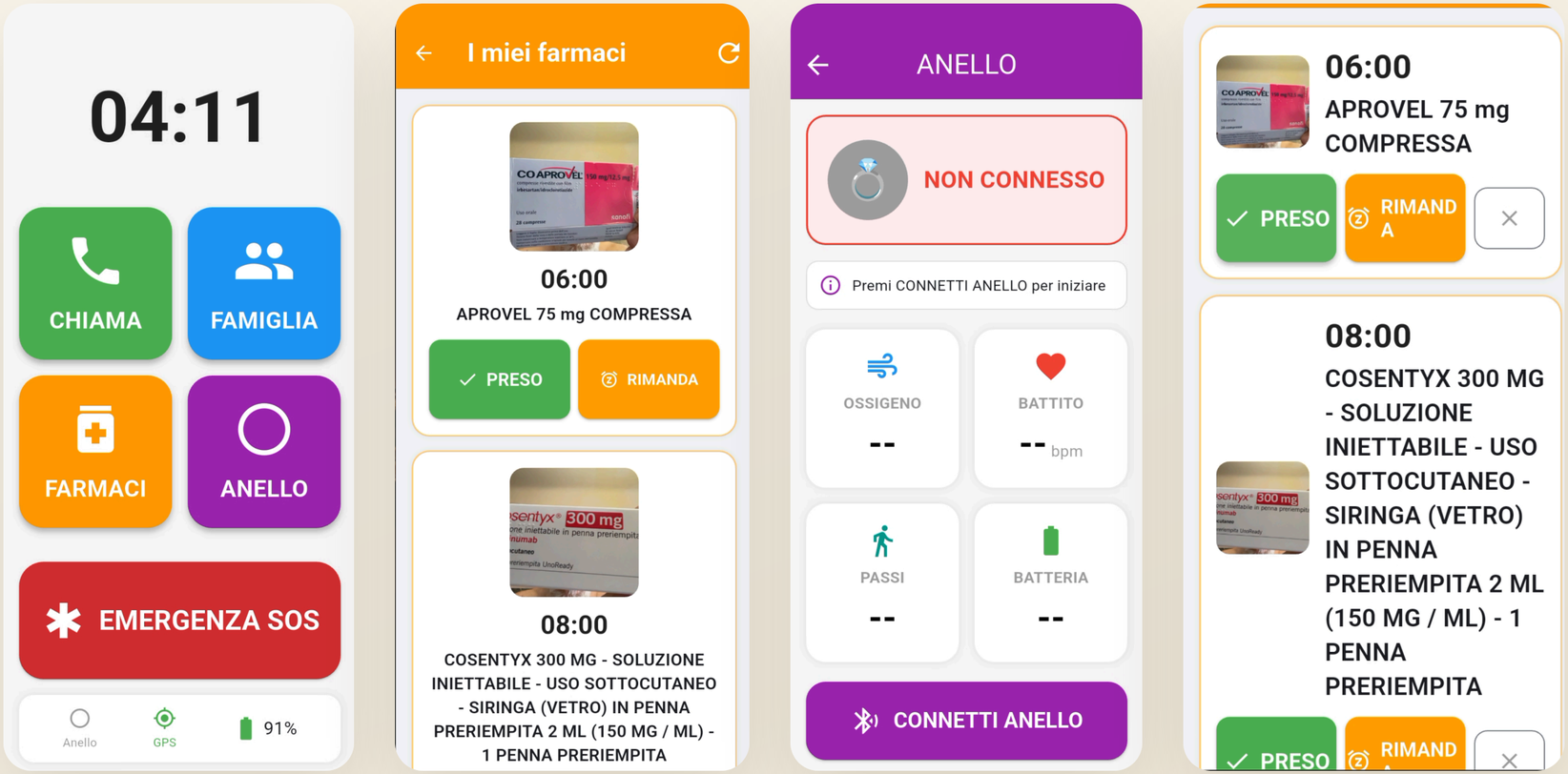
3 **Salvataggio in cloud**

La misura viene storicizzata e, in automatico, valutata dal motore di alert.

★ **Dispositivo dedicato, sistema bloccato**

Sistema operativo semplificato: all'avvio parte **solo e sempre** la nostra applicazione, unica utilizzabile. Comandi e interfaccia adattati al paziente anziano, touch target ≥ 44px.

L’app, così com’è oggi. Si apre e rimane bloccata.



09 — FLUSSO UTENTE · CONTROL ROOM

Un operatore, molti pazienti, una sola schermata.

Cruscotto in tempo reale

Pazienti attivi, dispositivi connessi, misurazioni odierne, alert delle ultime 24 ore.

Alert per gravità

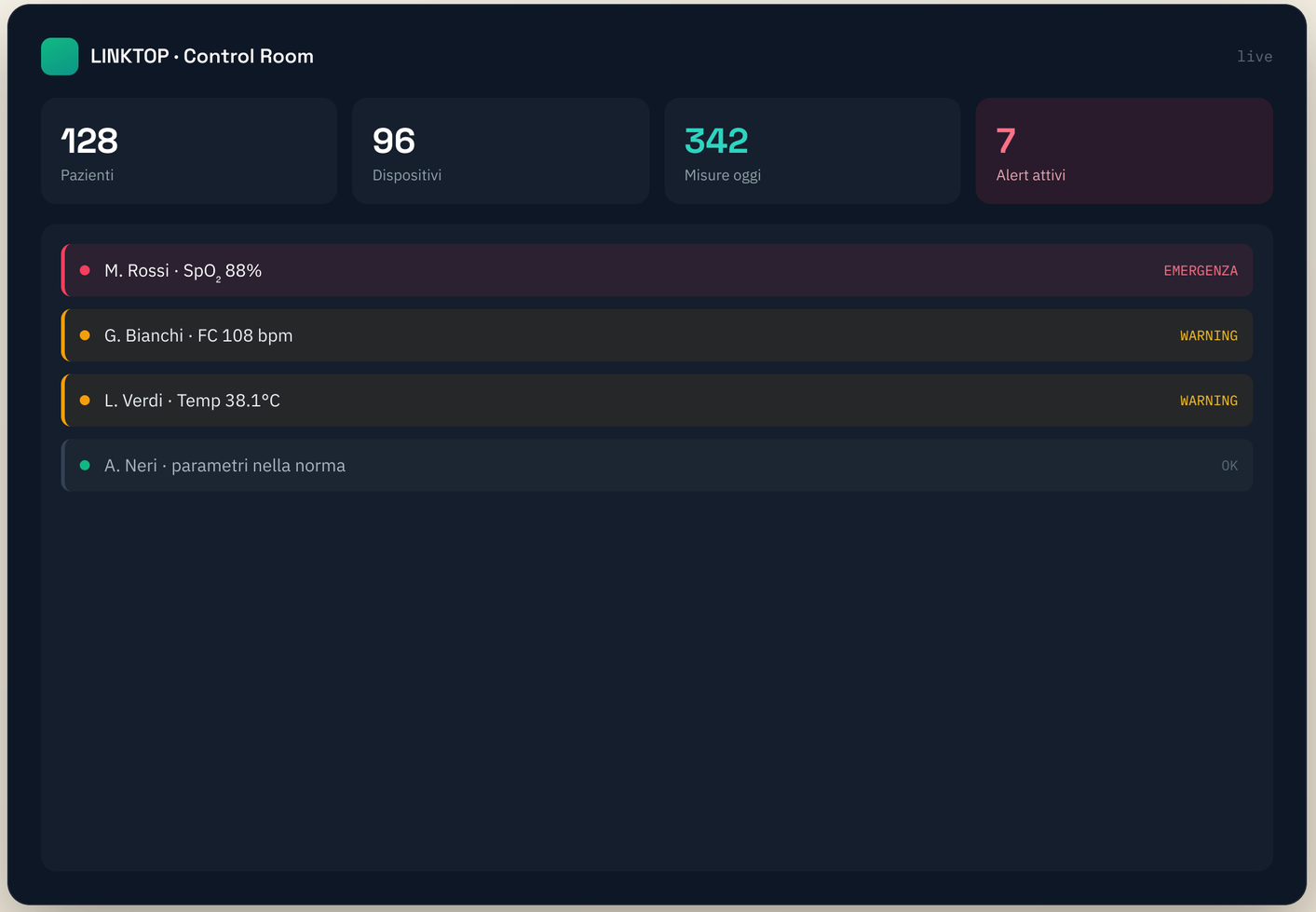
Coda ordinata: emergenze in cima, poi allarmi e warning. Acknowledge e risoluzione tracciati.

Dettaglio paziente

Grafici dei trend, storico misure, posizione e cronologia GPS su mappa.

Modello flessibile

Non serve necessariamente un infermiere di presidio: la stessa control room può essere il **cruscotto del medico**, che dal proprio studio vede in cloud i valori dei **soli pazienti** assegnati a lui.



PARTE II · IL MOTORE CLINICO

Il sistema di alert automatico.

Dove i dati grezzi diventano decisioni: valutazione delle soglie, prevenzione dei falsi allarmi, notifica ai contatti giusti.

10 — ALERT ENGINE

Cosa succede a ogni misurazione.

01 Recupera le **soglie effettive** del paziente

02 Valuta i 5 parametri — prima i **critici**, poi i **warning**

03 Anti-duplicati: ignora se c'è già un alert recente (5′ critico · 15′ warning)

04 Genera tipo + **messaggio descrittivo** e salva l'alert

05 Dispatch **asincrono** delle notifiche — non blocca la misura

```
// alerts/engine.ts
if (value < threshold.min_critical) {
severity: 'emergency',
direction: 'low'
}

// poi warning…
if (value > threshold.max_warning) {
severity: 'warning'
}

// messaggio generato
"CRITICO: Saturazione
ossigeno sotto la soglia.
Valore: 88% (soglia: 92%)"
```

11 — SOGLIE PERSONALIZZATE

Soglie cliniche su misura del singolo paziente.

BASE

Regole globali

Valori di riferimento validi per tutti, in cache 5 minuti.

→

OVERRIDE

Regole paziente

Soglie specifiche per il singolo, sovrascrivono le globali.

→

PRIORITÀ MAX

Custom JSONB

Soglie ad hoc nel profilo paziente, hanno l'ultima parola.

`getEffectiveThresholds() → merge: global ▸ patient ▸ custom`

12 — NOTIFICHE & DISPATCHER

L’allerta arriva alle persone giuste.

Email automatiche

Inviate via Resend con template dedicato per severità, valore e link alla dashboard.

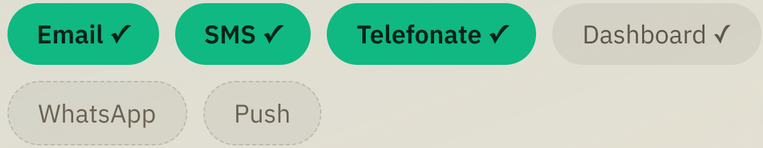
Tracciamento di stato

Ogni notifica è un record: `pending → sent → delivered / failed`, con id provider ed errori.

Escalation & destinatari

Caregiver e fino a due contatti di emergenza per paziente, con preferenze e quiet hours.

CANALI DI NOTIFICA



Email, SMS e telefonate preregistrate raggiungono i destinatari giusti; WhatsApp e push si attivano senza riprogettare il flusso.

13 — CONTROLLO ACCESSI

Cinque livelli di permessi, una sola fonte di verità.

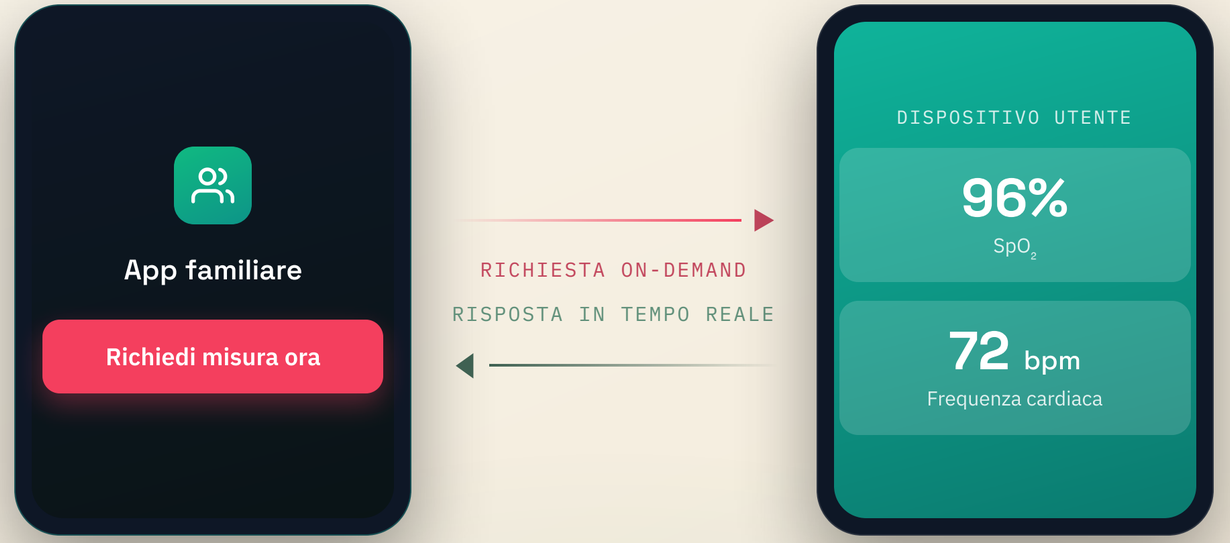
L4	**Sviluppatore**	Accesso completo · gestione utenti · log di sistema
L3	**Animatore digitale**	Configurazione dispositivi medici · soglie alert
L2	**Controllo parentale**	Monitoraggio pazienti assegnati · alert di emergenza
L1	**Utente base**	Visualizzazione dei propri dati

★ DIFFERENZIATORE CHIAVE CONTROLLO ATTIVO, NON SOLO MONITORAGGIO

Il familiare interroga il dispositivo dell’assistito, in qualsiasi momento.

Dalla propria app, con un tocco, chiede una misura **on-demand** — e riceve subito **saturazione** e **frequenza cardiaca**. Non aspetta che i dati arrivino: li va a prendere.

Richiede che l'assistito indossi il **dispositivo indossabile** e abbia con sé lo **smartphone**.



14 — APPLICAZIONE MOBILE

Un’app Flutter dedicata agli anziani.

elderly_monitoring_app — monitoraggio continuo, posizione e contatto, anche a schermo spento.

Tracking continuo

Foreground service che rileva parametri e posizione 24/7, con wakelock e riavvio automatico al boot.

`flutter_foreground_task`

Dispositivo indossabile

Connessione BLE al dispositivo indossabile per HR, SpO_2 e attività, sincronizzati con il cloud.

`flutter_blue_plus`

Videochiamata

Contatto diretto col caregiver tramite videochiamata integrata, senza app esterne.

`Jitsi / JaaS`

GPS & posizione

Localizzazione con storico e richiesta di posizione on-demand dalla control room.

`geolocator · geocoding`

Notifiche locali

Promemoria e avvisi direttamente sul dispositivo dell'assistito.

`local_notifications`

Sincronizzazione

Stesso backend Supabase della piattaforma web: un solo cloud, due interfacce.

`supabase_flutter`

15 — GEOFENCING

Zone sicure per chi tende a smarrirsi.

Zona ”Casa”

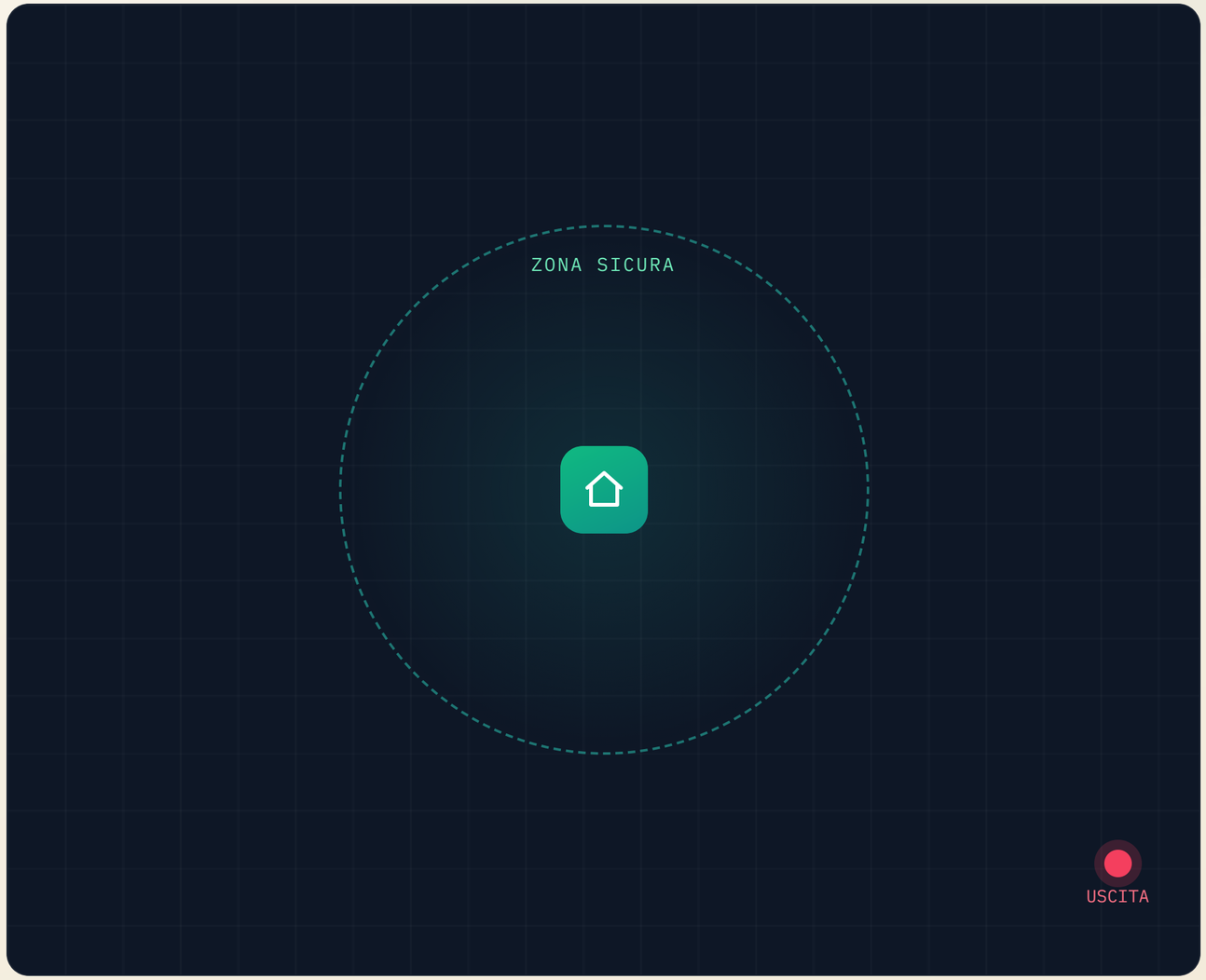
Si definisce un'area sicura centrata sull'abitazione o sulla struttura.

Controllo posizione

L'app confronta in continuo le coordinate con i confini delle zone attive.

Alert di uscita

Se l'assistito esce dall'area, un alert geofence avvisa subito i familiari, con la posizione.

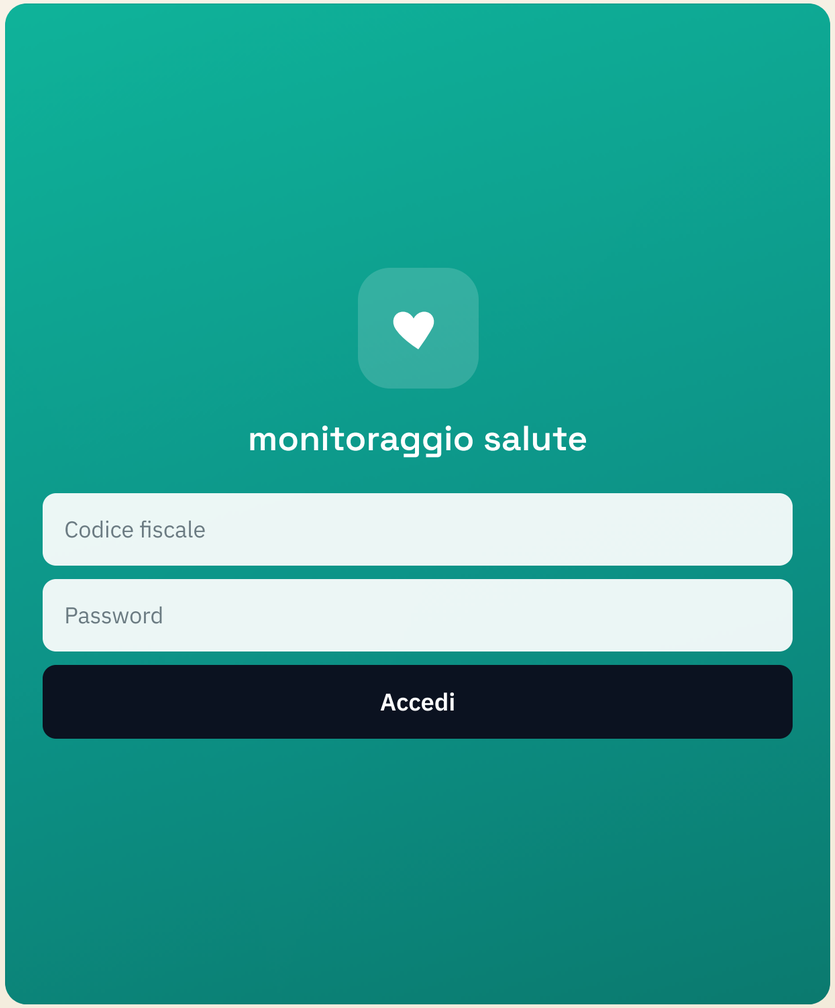


16 — SCHERMATE DEL PRODOTTO

Dal cloud al domicilio del paziente.



Dashboard Control Room — web



Accesso paziente

17 — SICUREZZA & DATI

Dati sanitari trattati con criterio.

Autenticazione

Password con hash `bcrypt` e sessioni firmate `JWT`.

Permessi server-side

Il controllo dei ruoli è applicato nel backend, non solo nascondendo voci di menu.

Log degli accessi

Tracciamento delle azioni e degli accessi, consultabile dallo sviluppatore.

Consensi privacy

Documenti di consenso e consegna generati e archiviati per ciascun paziente.

Validazione CF

Controllo formale del codice fiscale come identificativo del paziente.

Anti-duplicati alert

Finestre temporali che evitano allarmi ripetuti e "allarme da fatica" negli operatori.

Pensata per il cloud, senza rinunciare alla continuità.

Deploy serverless

Vercel: build automatica da Git, scaling gestito, latenza ridotta scegliendo la stessa region del DB.

Database gestito

Supabase (PostgreSQL) con `Session Pooler` IPv4 e SSL — connessione affidabile da Vercel.

PWA & health-check

App installabile con service worker; monitoraggio di disponibilità automatico via GitHub Actions.

Architettura ibrida

Configurazione che supporta sia il cloud Supabase sia la VPS legacy — migrazione graduale, zero downtime.

19 — IL PROGETTO IN NUMERI

Un sistema completo, non un prototipo.

2

applicazioni full-stack

`Web (Next.js) + Mobile (Flutter)`

37+

endpoint API serverless

`auth · health · alert · gps · video`

4

dispositivi integrati

`Smartphone dedicato Kiosko - Anello`

5

parametri vitali valutati

`FC · SpO₂`

5

livelli di ruolo (RBAC)

`da utente base a sviluppatore`

3

livelli di soglie alert

`globali ▸ paziente ▸ custom`

Quattro padronanze in un solo prodotto.

A

Full-stack moderno

Next.js 16 / React 19, TypeScript end-to-end, API serverless e PostgreSQL ben modellato.

B

Integrazione hardware

Web Bluetooth e BLE nativo, con protocolli di dispositivi decodificati per reverse engineering.

C

Logica di controllo

Motore di alert con soglie gerarchiche, severità, escalation e notifiche tracciate.

D

Cloud & operatività reale

Deploy serverless, DB gestito, PWA, kiosk in RSA: già installato e funzionante sul campo.

IL PRODOTTO ESISTE, FUNZIONA, È SUL CAMPO

Dalla misura all’allerta, automaticamente.

Una piattaforma di telemonitoraggio completa — web, mobile, hardware e cloud — pronta a scalare verso nuovi pazienti, strutture e canali di allerta.