



UNIVERSITÀ
DEL SALENTO

Smart Heart Monitoring

Analisi e Prevenzione delle Anomalie Cardiache
con AI e IoT

SVILUPPO STORICO



AI e il Futuro del Monitoraggio Sanitario



2020-oggi: Intelligenza Artificiale e Medicina Personalizzata

Gli smartwatch di ultima generazione utilizzano machine learning per rilevare aritmie, ipertensione e anomalie in tempo reale. Grazie a sensori avanzati, monitorano biomarcatori come glucosio e cortisolo, supportando la diagnosi predittiva di malattie croniche e favorendo interventi tempestivi.



OBIETTIVO DEL PROGETTO



Il progetto mira a sviluppare un sistema avanzato di monitoraggio della frequenza cardiaca basato su intelligenza artificiale e dispositivi wearable, in grado di identificare anomalie cardiache e notificare l'utente in tempo reale.

REQUISITI FUNZIONALI

1 Monitoraggio della Frequenza Cardiaca

Acquisizione dati da MongoDB, selezione per data e download in formato CSV.

2 Rilevamento delle Anomalie

Uso di Isolation Forest per individuare anomalie, con regolazione della sensibilità e visualizzazione nei grafici.



3 Visualizzazione Interattiva

Dashboard con grafici dinamici (linea, scatter plot, istogramma) e filtri temporali.

4 Notifiche Telegram

Avvisi automatici in caso di anomalie critiche, con dettagli su data e numero di eventi.

5 Accessibilità e Usabilità

Dashboard web-based, intuitiva e accessibile da qualsiasi browser senza installazioni.

REQUISITI NON FUNZIONALI

- 1 Prestazioni e Scalabilità**
Tempo di risposta , gestione dataset senza perdita di performance.
- 2 Sicurezza e Privacy**
Dati accessibili solo all'utente, protezione delle credenziali API, autenticazione sicura per MongoDB.
- 3 Portabilità e Compatibilità**
Eseguibile su qualsiasi browser, compatibile con Windows, macOS, Linux e versioni recenti di Python e Streamlit.
- 4 Manutenibilità ed Estendibilità**
Codice modulare, supporto per nuove funzionalità e integrazione con API di smartwatch.
- 5 Affidabilità**
Gestione errori di connessione a MongoDB, messaggi chiari in caso di dati mancanti, nessun crash.

TECNOLOGIE UTILIZZATE NEL SISTEMA

Smartwatch – Google Pixel Watch 2

Dispositivo con monitoraggio continuo della frequenza cardiaca 24/7, elevata precisione e compatibilità con API Fitbit per l'acquisizione dati.



MongoDB

Database NoSQL scalabile, ideale per gestire grandi volumi di dati sanitari con accesso rapido e flessibile.



API Telegram

Sistema di alert in tempo reale, che invia notifiche automatiche agli utenti in caso di anomalie cardiache.



Python

Linguaggio scelto per la sua versatilità, con librerie per analisi dati (Pandas), Machine Learning (Scikit-Learn), database (PyMongo) e API.

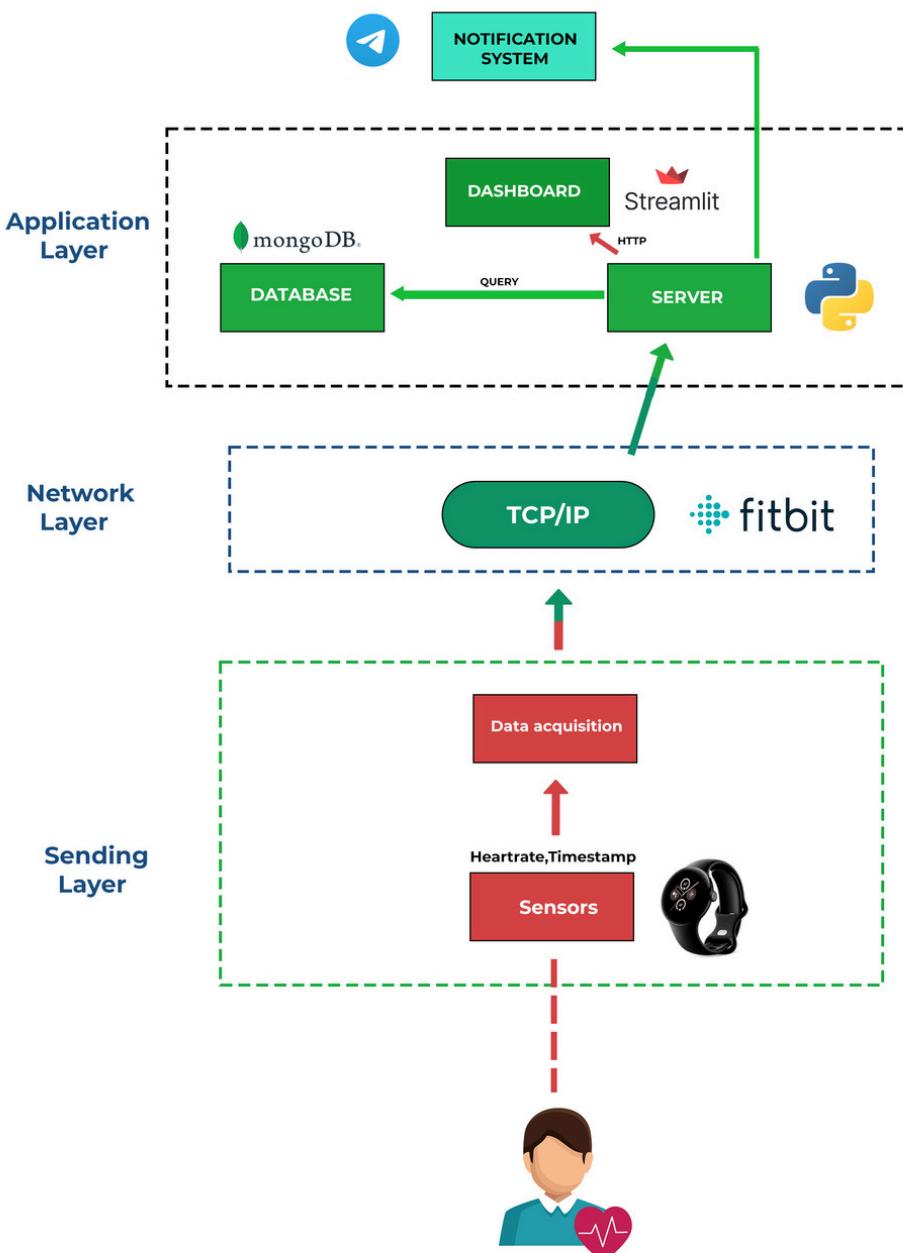


Streamlit

Framework Python per creare una dashboard interattiva, con grafici dinamici, filtri temporali e gestione delle anomalie.



ARCHITETTURA DI SISTEMA



STRUTTURA DEL DATABASE

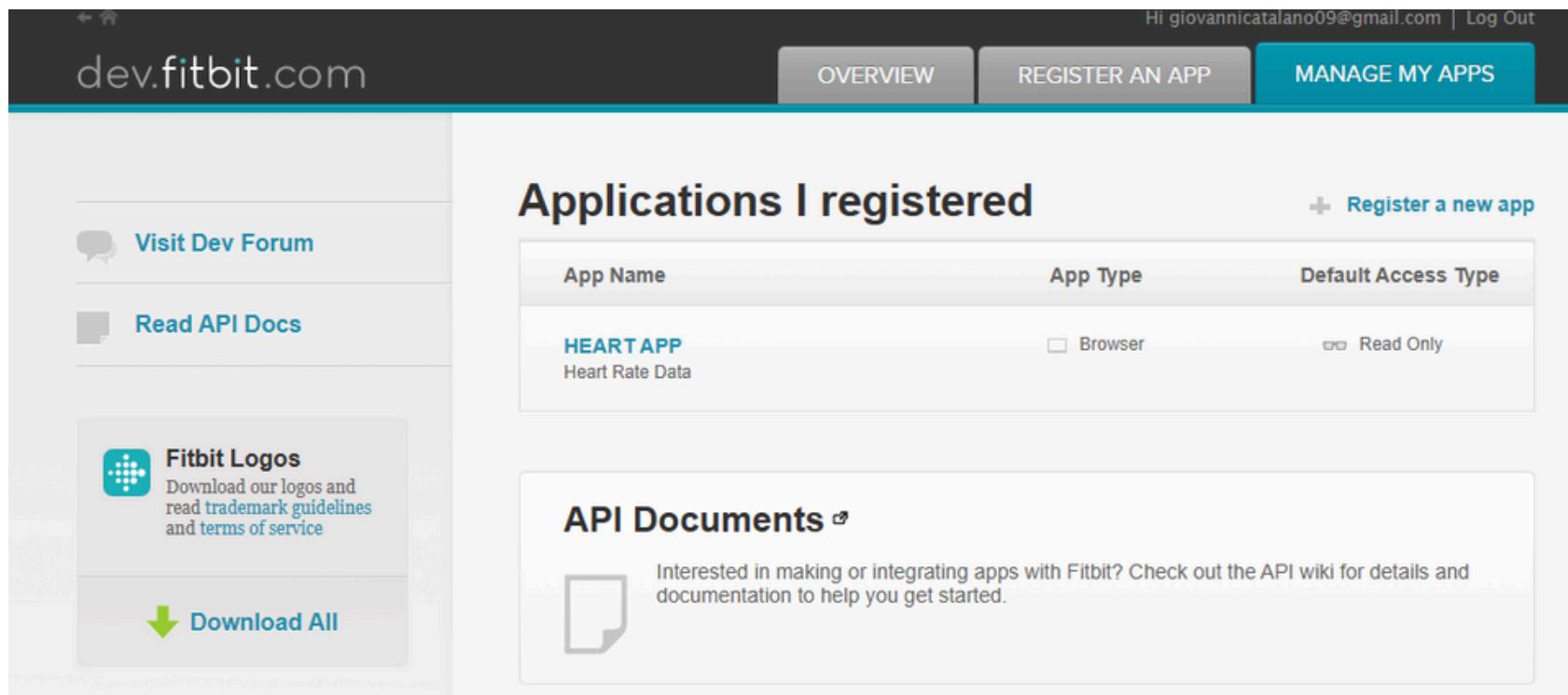
heart_rates

Storage size: 892.93 kB	Documents: 49 K	Avg. document size: 59.00 B	Indexes: 1	Total index size: 528.38 kB
-----------------------------------	---------------------------	---------------------------------------	----------------------	---------------------------------------

utente

Storage size: 20.48 kB	Documents: 1	Avg. document size: 226.00 B	Indexes: 1	Total index size: 20.48 kB
----------------------------------	------------------------	--	----------------------	--------------------------------------

SICUREZZA CON O-AUTH



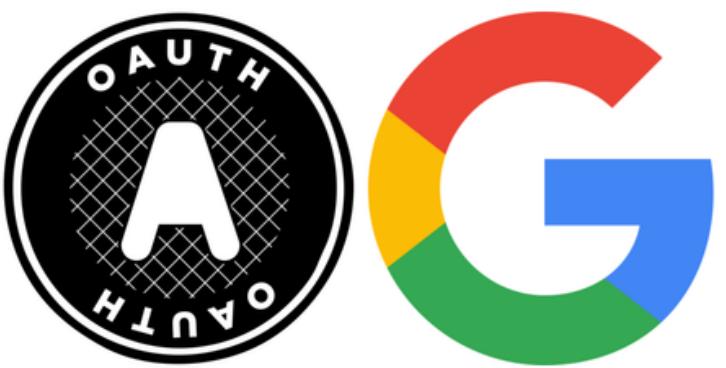
The screenshot shows the 'Applications I registered' section of the Fitbit API developer portal. At the top, there are three buttons: 'OVERVIEW', 'REGISTER AN APP', and 'MANAGE MY APPS'. The 'MANAGE MY APPS' button is highlighted in blue. The main table lists one application: 'HEART APP' (Heart Rate Data), which is a 'Browser' type app with 'Read Only' access. To the right of the table is a link to 'Register a new app'. On the left sidebar, there are links to 'Visit Dev Forum', 'Read API Docs', 'Fitbit Logos' (with download options), and a 'Download All' button.

Accesso alle API Fitbit tramite OAuth 2.0 con Google

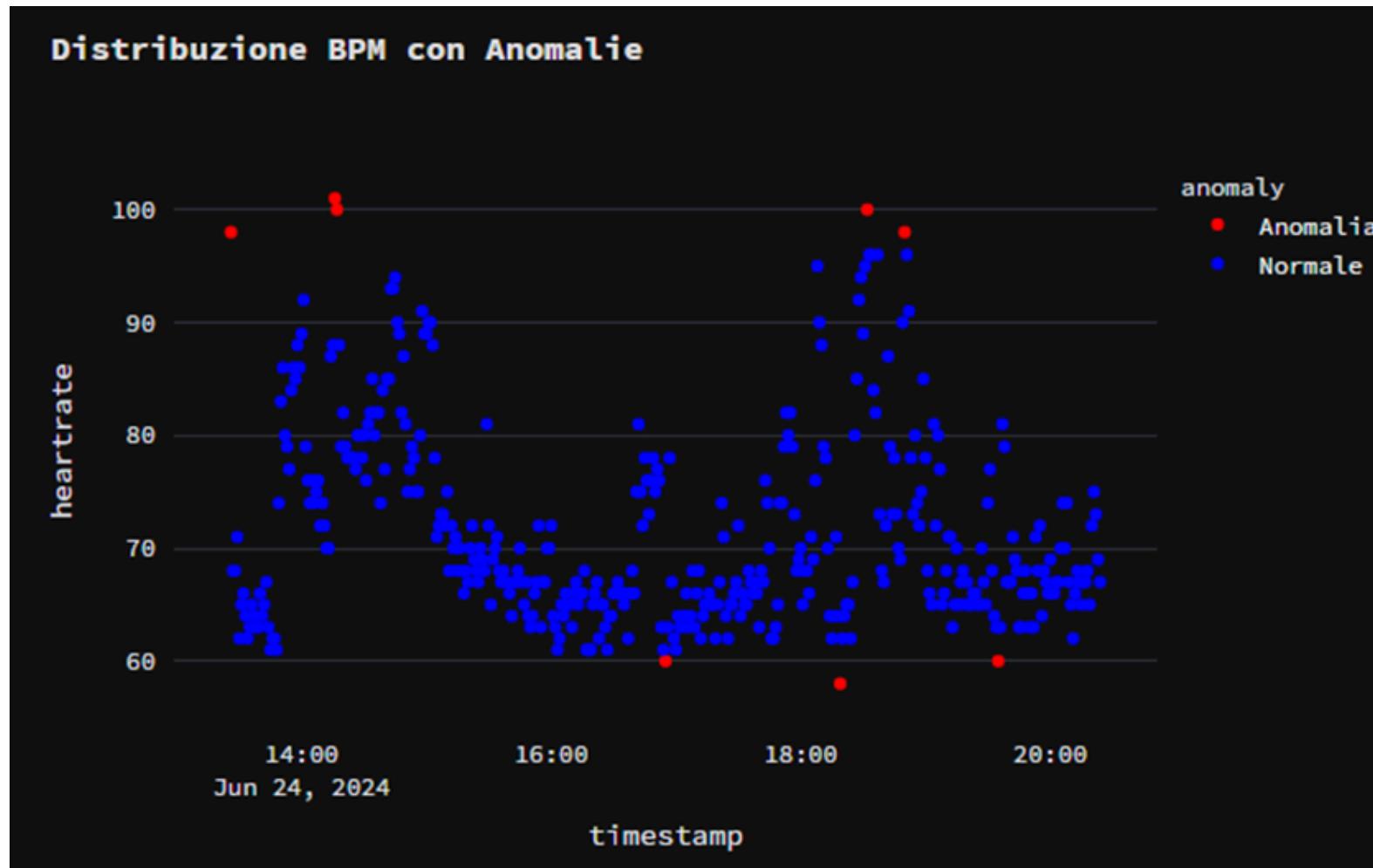
Grazie all'implementazione del protocollo OAuth 2.0 con autenticazione tramite Google, è stato possibile accedere in modo sicuro alle **API web di Fitbit**.

L'utente effettua il login utilizzando il proprio account Google e, attraverso un processo di autorizzazione esplicita, concede i permessi necessari per consentire all'applicazione di accedere ai dati personali registrati da Fitbit, come la frequenza cardiaca, il numero di passi o le fasi del sonno.

Una volta ottenuto l'access token, è possibile effettuare richieste alle **API Fitbit** e ricevere dati aggiornati in tempo reale, il tutto senza dover gestire manualmente credenziali o dati sensibili. Questo approccio garantisce un alto livello di sicurezza, tutela della privacy e una migliore esperienza per l'utente.



MODELLO ISOLATION FOREST



Per l'individuazione di anomalie nei dati di frequenza cardiaca ho utilizzato il modello di machine learning Isolation Forest, una tecnica non supervisionata progettata appositamente per rilevare outlier

A differenza di altri modelli, Isolation Forest si basa sul principio che i punti anomali sono più facili da isolare rispetto ai punti normali. Il modello costruisce molteplici alberi binari che suddividono i dati in modo casuale; gli elementi che vengono isolati con meno suddivisioni sono considerati potenziali anomalie.

Applicando questo approccio ai dati raccolti tramite le API di Fitbit, il modello è stato in grado di identificare automaticamente valori di frequenza cardiaca anomali rispetto al comportamento fisiologico abituale dell'utente, segnalando variazioni sospette che potrebbero indicare irregolarità o eventi significativi.

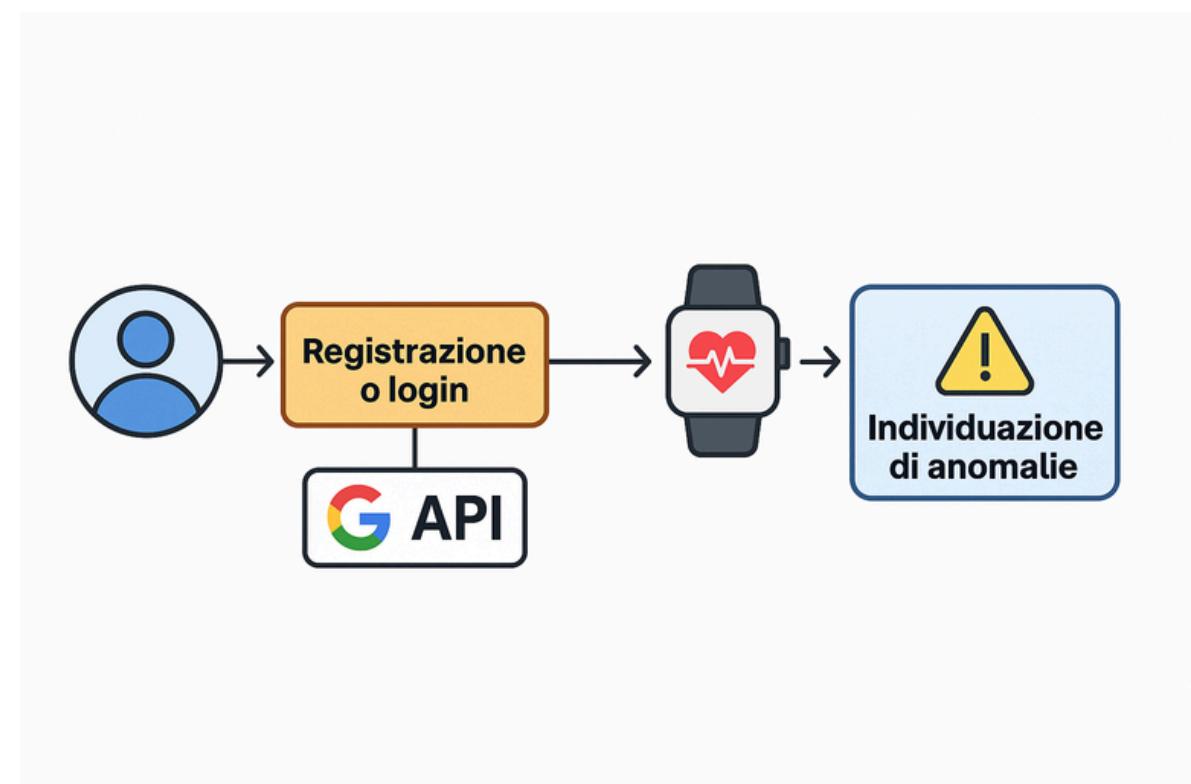
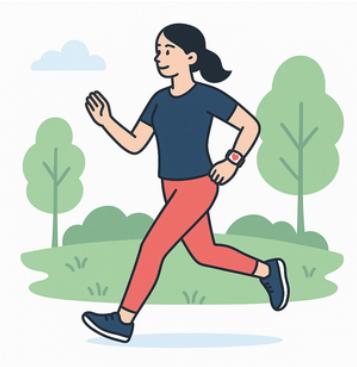
IMPLEMENTAZIONE DEL MODELLO



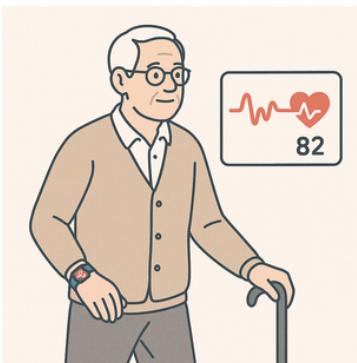
Il modello di rilevamento delle anomalie è stato implementato utilizzando la libreria scikit-learn, tramite il modulo IsolationForest importato da sklearn.ensemble. Dopo aver raccolto i dati della frequenza cardiaca attraverso le API Fitbit e averli organizzati in un DataFrame Pandas, ho eseguito una fase di pulizia e normalizzazione dei valori. Successivamente, ho istanziato il modello IsolationForest specificando il numero di estimatori e il tasso di contaminazione previsto, ovvero la percentuale attesa di outlier. Una volta addestrato il modello sui dati, ho utilizzato il metodo predict() per classificare ogni osservazione come normale o anomala, e il metodo decision_function() per misurare la distanza di ciascun punto dalla frontiera di anomalia. I risultati ottenuti sono stati poi visualizzati in Streamlit, evidenziando i valori anomali su grafico interattivo tramite Plotly.

ESEMPIO DI UTILIZZO

Un caso d'uso tipico riguarda un utente che si registra tramite autenticazione Google e collega il proprio account Fitbit per monitorare la frequenza cardiaca durante l'attività sportiva. Una volta autorizzato l'accesso ai dati, il sistema raccoglie in tempo reale le misurazioni del battito cardiaco e applica il modello Isolation Forest per rilevare eventuali anomalie. Ad esempio, durante una sessione di corsa, il modello può distinguere tra normali aumenti della frequenza cardiaca e picchi inaspettati che potrebbero indicare un sovraccarico o un potenziale rischio fisico.



In ambito clinico, invece, un paziente ricoverato o seguito da remoto può essere monitorato costantemente attraverso lo stesso sistema. In questo caso, l'obiettivo è intercettare tempestivamente variazioni anomale nella frequenza cardiaca a riposo, come tachicardie improvvise o irregolarità potenzialmente critiche. Quando viene rilevato un outlier, il sistema può inviare automaticamente un alert (es. tramite Telegram) al personale sanitario o al paziente stesso, permettendo un intervento rapido e mirato.



SVILUPPI FUTURI



- **Ottimizzazione del modello di rilevamento**
Verranno migliorati i parametri e le tecniche di addestramento del modello per aumentare la precisione nell'identificazione delle anomalie, riducendo sia i falsi positivi che i falsi negativi.
- **Implementazione locale su dispositivi indossabili**
L'obiettivo è portare il modello di rilevamento direttamente sullo smartwatch o sul dispositivo mobile, permettendo un'analisi in tempo reale anche offline, con maggiore efficienza e minor dipendenza da server esterni.
- **Estensione ai parametri biometrici aggiuntivi**
Si prevede di includere nuove metriche fisiologiche come la variabilità della frequenza cardiaca (HRV), la saturazione dell'ossigeno nel sangue (SpO_2) e la qualità del sonno, al fine di offrire un quadro più completo della salute dell'utente.

CONCLUSIONE

GRAZIE PER L'ATTENZIONE

