

Sorveglianza Agnostica Multivariabile



Framework di analisi composita

Copyright 2026

Tutti i diritti riservati

In copertina e nella premessa le immagini sono generate da AI/

Release documento 260203



260203

Indice

Presentazione Digitale

Premessa Letteraria

1. Score di scostamento

- 1.1 Esempio di scores da PWA IceSense
- 1.2 Differenza di media (Z-score)
- 1.3 Trend (pendenza)
- 1.4 Varianza (stabilità)
- 1.5 Forma della curva (RMSE)
- 1.6 Calcolo dello score finale
- 1.7 Modello di determinazione della baseline
- 1.8 Granularità dei dati e scalabilità degli score
- 1.9 Tempo di refresh dello score

2. Rilevamento deterministico delle anomalie

- 2.1 Invio dati all'assistenza tecnica
- 2.2 Invio dati all'AI
- 2.21 Richiesta API DeepSeek
- 2.22 Formato del prompt
- 2.23 Snippet di guasto filtrato
- 2.24 Risposta reale dell'AI
- 2.25 Costi

3. Sorveglianza complementare: Snapshot System

- 3.1 Funzioni del modello PWA IceSense
- Esempio di snapshot restituito dal controllore

4. Sorveglianza complementare: Snapshot Functions

- Esempio di snapshot restituito dal controllore

Premessa Digitale



Inquadra il QR per un'anteprima
multimediale del contenuto

Premessa Letteraria

Sorveglianza agностica

La sorveglianza agnostica multivariabile nasce dall'esigenza di analizzare il comportamento reale dell'impianto senza ricorrere a soglie fisse, regole predefinite o modelli dipendenti dall'impianto. Nei sistemi fisici complessi non è possibile definire limiti universali o prevedere in modo deterministico l'insorgere di un guasto: ciò che invece è sempre osservabile è l'allontanamento progressivo di una variabile dal proprio comportamento storico. Il modello adottato si basa esattamente su questo principio: misurare lo scostamento dalla normalità statistica, non predire il guasto.

Lo score di scostamento fornisce un indicatore quantitativo, stabile e riproducibile dell'evoluzione delle grandezze monitorate, indipendentemente dal tipo di impianto, dalla variabile e dal contesto operativo.

Analisi deterministica

A differenza della sorveglianza agnostica, l'analisi deterministica estrae in tempo reale gli eventi di guasto, allerta i soggetti operativi, e se l'unità è connessa invia al cloud lo stato di allarme. Ogni evento deterministico è quindi progettato per raggiungere rapidamente chi deve intervenire sul campo.

Ruolo degli snapshot

Tuttavia, uno scostamento numerico o uno stato di allarme non è sufficiente per avere un'immagine completa di tutte le funzionalità: serve un contesto tecnico che descriva lo stato del controllore, del firmware, e dell'applicazione. Per questo motivo il sistema integra un secondo elemento di analisi: gli snapshot, ovvero report tecnici strutturati che raccolgono e organizzano le informazioni necessarie per interpretare il funzionamento globale. Analisi agnostica, deterministica e snapshot costituiscono quindi tre livelli complementari.

Ruolo dell'AI

A questi tre livelli si aggiunge l'intelligenza artificiale, che non esegue calcoli tecnici ma svolge una funzione di interpretazione linguistica. L'AI riceve un prompt strutturato, derivato dai dati filtrati e consolidati, e restituisce una spiegazione chiara e coerente delle anomalie rilevate, utile soprattutto per l'utente non tecnico. L'AI non sostituisce la diagnostica, ma la rende leggibile e immediatamente utilizzabile.

Ciclo informativo dell'ecosistema

Il sistema di sorveglianza nel suo complesso crea un linguaggio tecnico comune tra tutte le competenze coinvolte: l'utilizzatore sul campo, lo sviluppatore software, chi scrive il firmware, chi progetta l'hardware e il termodinamico che interpreta le grandezze fisiche.

Ogni elaborato parte dal dispositivo fisico, passa attraverso l'utente tramite connessione NFC o tramite cloud, raggiunge le competenze software, firmware, hardware e termodinamiche, e ritorna sul campo sotto forma di interventi mirati. Il sistema non si limita a descrivere un singolo evento, ma contribuisce a individuare pattern, condizioni ricorrenti e anomalie utili al miglioramento continuo dell'intera flotta. L'elaborazione avviene interamente sul controllore. Per le unità connesse il cloud riceve solo indicatori già calcolati, evitando qualsiasi carico computazionale. La piattaforma funge da vettore di distribuzione e correlazione, non da motore di analisi.

1. Score di scostamento

Un vantaggio fondamentale dell'approccio adottato è la totale agnoscibilità del modello:

- non richiede configurazioni specifiche per l'impianto,
- non dipende da limiti preimpostati,
- si adatta a qualunque variabile fisica o elettrica, incluse quelle ricavate dal monitoraggio della MCU del controllore,
- mantiene coerenza e riproducibilità indipendentemente dal contesto operativo.

Lo score non produce allarmi, ma segnali di attenzione: un indicatore oggettivo, stabile e trasversale, che permette di concentrare l'analisi sulle variabili realmente significative, migliorando la capacità di diagnosi precoce e riducendo il rischio di interpretazioni soggettive.

Lo score è calcolato attraverso la somma di quattro controlli diversi, ciascuno progettato per individuare un tipo specifico di cambiamento nei dati analizzati. Questo approccio permette di identificare anomalie o variazioni rilevanti rispetto a una baseline storica.

Per il modello concettuale (PWA "IceSense") si è scelto un campionamento orario giornaliero a bassa granularità (1 campione ogni ora, pari a 24 campioni al giorno).

1.1 Esempio scores da PWA "IceSense"

La variabile con maggior punteggio è il condensatore; per il punteggio restituito (circa 70) va attenzionata; successivamente si potrebbe presentare un guasto sullo scambio termico.



1.2. Differenza di media (Z-score)

Il primo controllo valuta quanto la media dei dati attuali si discosta dalla media della baseline. Maggiore è la differenza tra le due medie rispetto alla variabilità storica, più aumenta il contributo allo score. Il valore massimo attribuibile è pari a 100.

Formula:

$$Z = (\mu_{\text{current}} - \mu_{\text{baseline}}) / \sigma_{\text{baseline}}$$

Variabili:

- μ_{current} : media degli n valori attuali
- μ_{baseline} : media degli n valori della baseline
- σ_{baseline} : deviazione standard degli n valori della baseline

1.3. Trend (pendenza)

Il secondo controllo verifica se negli ultimi N punti esiste una tendenza chiara e significativa (in crescita o in diminuzione). La pendenza contribuisce allo score solo se supera una soglia minima, evitando così falsi allarmi. Più la pendenza è marcata, maggiore sarà il contributo.

Formula della pendenza:

$$\text{slope} = [N \cdot \sum(x_i \cdot y_i) - (\sum x_i)(\sum y_i)] / [N \cdot \sum(x_i^2) - (\sum x_i)^2]$$

Variabili:

- N = numero dei campioni
- x_i : indice del punto (da 1 a N)
- y_i : valore corrente al punto i

1.4. Varianza (stabilità)

Il terzo controllo confronta la variabilità dei dati attuali con quella della baseline. Se la variabilità attuale risulta troppo superiore o inferiore rispetto al "normale", viene aggiunto un contributo allo score. Se la variabilità rientra nell'intervallo considerato normale, non influisce sullo score.

Formula: $R = \sigma_{\text{current}} / \sigma_{\text{baseline}}$

Variabili:

- σ_{current} : deviazione standard degli n valori attuali
- σ_{baseline} : deviazione standard degli n valori della baseline

Condizioni di attivazione: $R < 0.4$ oppure $R > 2.5$

Scostamento considerato: $|R - 1|$

Nota importante

La valutazione della variabilità non utilizza limiti operativi né soglie legate all'impianto, ma si basa esclusivamente sulla coerenza statistica tra il comportamento attuale e quello storico. La dinamica è considerata "normale" quando la variabilità rimane vicina a quella della baseline; quando invece si riduce in modo marcato o aumenta in maniera significativa, il sistema interpreta questo cambiamento come un'evoluzione non più compatibile con la normalità.

I valori 0.4 e 2.5 non rappresentano soglie fisiche o parametri da configurare, ma semplicemente l'intervallo entro cui la variabilità può essere considerata statisticamente coerente. Al di sotto di 0.4 la dispersione si riduce a meno della metà, mentre oltre 2.5 risulta più che raddoppiata: in entrambi i casi la dinamica del segnale cambia natura e questo cambiamento contribuisce allo score.

Il contributo non deriva dal superamento di un limite, ma dallo scostamento dalla normalità, espresso in forma simmetrica come:

$$| R-1 |$$

In questo modo il modello rimane completamente agnostico: non interpreta questi valori come allarmi o soglie operative, ma come semplici indicatori matematici di quanto la variabilità attuale si discosti da quella storica, indipendentemente dal tipo di variabile o dall'impianto.

1.5. Forma della curva (RMSE)

Il quarto controllo confronta punto per punto la forma delle due curve (baseline e attuale), calcolando quanto la forma complessiva è cambiata. Un maggiore scostamento della curva attuale rispetto alla baseline aumenta lo score.

Formula RMSE:

$$\text{RMSE} = \sqrt{[(1 / N) \cdot \sum (y_{i,\text{current}} - y_{i,\text{baseline}})^2]}$$

Normalizzazione: $\text{RMSE norm} = \text{RMSE} / \sigma_{\text{baseline}}$

Variabili:

- $y_{i,\text{current}}$: valore corrente al punto i
- $y_{i,\text{baseline}}$: valore baseline al punto i
- N = numero dei campioni
- σ_{baseline} : deviazione standard della baseline

1.6. Calcolo dello Score Finale

Ogni controllo statistico produce un contributo proporzionale allo scostamento rilevato, ma ciascun contributo viene pesato e poi limitato così che nessun singolo fenomeno possa saturare lo score. La somma dei contributi è a sua volta vincolata a un massimo di 100, mantenendo lo score sempre stabile e confrontabile. Un ammorbidente finale riduce la sensibilità ai picchi e rende la crescita più progressiva, così lo score riflette l'allontanamento dalla normalità senza diventare nervoso o eccessivamente reattivo.

Formula finale:

$$SCO = (\text{score grezzo} / 100)^{0.55} \times 100$$

La baseline mantiene sempre gli N valori originali, essenziali per i confronti futuri e per la coerenza del metodo di analisi. L'esponente 0.55 è stato scelto empiricamente per ottenere una curva di crescita progressiva ma non piatta.

1.7. Modello di determinazione della baseline

La baseline viene generata automaticamente utilizzando solo i periodi di funzionamento stabile, esclusi aperture porta, sbrinamenti e stati speciali. Il controllore identifica queste condizioni in tempo reale, garantendo che la baseline rifletta esclusivamente il comportamento normale dell'impianto.

Una volta definita, la baseline non viene riscritta continuamente, ma aggiornata in modo lento e controllato: i nuovi dati vengono integrati solo quando il sistema si trova di nuovo in condizioni operative stabili, mentre i campioni acquisiti durante aperture porta, carichi di prodotto caldo o sbrinamenti sono esclusi dall'apprendimento. In questo modo la baseline segue gradualmente l'evoluzione naturale dell'impianto (stagionalità, invecchiamento), senza inseguire i disturbi operativi.

Quando lo score indica uno scostamento prolungato e significativo dalla normalità, la baseline viene mantenuta ferma e utilizzata come riferimento per la diagnosi, evitando che il modello si adatti a uno stato degradato. Dopo interventi strutturali sull'impianto o modifiche rilevanti, il sistema è in grado di ricostruire automaticamente una nuova baseline a partire dai nuovi periodi di funzionamento stabile, senza richiedere configurazioni manuali. In sintesi, la baseline non è un valore fissato a priori, ma un riferimento dinamico, appreso dal comportamento reale dell'impianto e continuamente mantenuto coerente grazie alle informazioni operative fornite dal controllore.

1.8. Granularità dei dati e scalabilità degli score

Nel modello concettuale usato per spiegare il funzionamento dello score, abbiamo adottato una rappresentazione semplificata basata su 24 campioni orari. È una scelta pensata per dimostrare la logica del metodo in modo immediato, senza appesantirlo con la complessità dei dati reali.

Nella pratica, però, i controllori utilizzano una granularità molto più fine: fino a 2.880 campioni al giorno, uno ogni 30 secondi. Ed è proprio in questo scenario ad alta densità informativa che il modello mostra la sua reale potenza.

Lo score non dipende dalla frequenza di campionamento, non richiede soglie fisse né regole specifiche per impianto, e non ha bisogno di tarature manuali. È un metodo completamente agnostico: si adatta automaticamente alla quantità di dati disponibili e sfrutta la granularità elevata per migliorare la qualità statistica delle medie, delle varianze, dei trend e della forma della curva.

In altre parole, più dati arrivano, più lo score diventa robusto, stabile e affidabile, senza diventare eccessivamente reattivo o sensibile al rumore. Questa caratteristica lo rende particolarmente efficace in due contesti operativi molto diversi.

Nel caso dei controllori che comunicano via NFC, dove ogni scambio è limitato a circa 1Kb, non è possibile trasferire grandi serie temporali. Lo score risolve il problema alla radice: tutta l'elaborazione avviene localmente e ciò che viene trasmesso è solo il risultato finale, leggero e compatibile con i vincoli fisici della tecnologia NFC. Il cliente riceve un'informazione sintetica ma estremamente significativa, senza saturare la banda.

All'estremo opposto, nei sistemi connessi al cloud, la granularità elevata potrebbe trasformarsi in un problema di carico computazionale se fosse la piattaforma a dover elaborare i dati di ogni dispositivo. Il nostro approccio evita questo rischio: il calcolo dello score avviene sul controllore, non nel cloud. La piattaforma riceve solo indicatori già elaborati, senza dover eseguire regressioni, RMSE o analisi statistiche per migliaia di device. Il risultato è una scalabilità naturale, con costi computazionali minimi e prestazioni costanti anche su flotte molto numerose.

La combinazione tra alta granularità reale, calcolo locale, output leggero e totale agnosticità rispetto alle grandezze fisiche ed elettriche, incluse quelle ricavate dalla MCU del controllore, rende lo score uno strumento affidabile, universale e industrialmente sostenibile.

Non predice il guasto, ma emula in modo credibile l'allontanamento dalla normalità, offrendo un vantaggio competitivo reale sia nei sistemi a bassissima banda sia nelle architetture cloud ad alta densità di dispositivi.

1.9 Tempo di refresh dello score

Nel modello concettuale è stato utilizzato un campionamento orario per semplificare la spiegazione del metodo. Nella pratica operativa, tuttavia, i controllori acquisiscono fino a 2.880 campioni al giorno (uno ogni 30 secondi). In questo contesto ad alta granularità, il tempo di refresh dello score assume un ruolo fondamentale per garantire stabilità, coerenza e robustezza statistica.

Lo score non viene ricalcolato a ogni campione: un aggiornamento così frequente renderebbe l'indicatore eccessivamente reattivo al rumore e alle oscillazioni fisiologiche del sistema. L'obiettivo del modello è invece misurare l'allontanamento progressivo dalla normalità, non le variazioni istantanee.

Per questo motivo il calcolo dello score avviene su una finestra mobile di N campioni (ad esempio 24 ore), confrontando la dinamica attuale con la baseline storica. L'aggiornamento dello score viene eseguito a intervalli regolari, tipicamente ogni 5–10 minuti, corrispondenti a 10–20 campioni. Questa frequenza rappresenta un equilibrio ottimale: sufficientemente rapida da cogliere evoluzioni significative, ma abbastanza lenta da filtrare il rumore e mantenere la stabilità dell'indicatore.

In sintesi, anche con una densità informativa elevata, il modello conserva la propria natura agnostica e robusta: la granularità fine migliora la qualità statistica dei controlli, mentre il refresh periodico evita reazioni impulsive e garantisce uno score affidabile, coerente e industrialmente utilizzabile e commercialmente onesto.

2. Rilevamento deterministico delle anomalie

Accanto al modello statistico degli score, il sistema integra un secondo criterio dedicato agli eventi improvvisi, cioè alle condizioni che richiedono una sorveglianza tecnica immediata (guasti effettivi). In questi casi, il controllo è deterministico, basato su limiti e range predefiniti. Ad ogni campionamento, il sistema verifica se i valori misurati rientrano nel range operativo previsto. Le variabili analizzate includono:

- Termostato: anomalia se il valore è fuori dal range previsto
- Temperatura uscita evaporatore: anomalia se il valore è fuori dal range previsto
- Temperatura uscita condensatore: anomalia se il valore supera la soglia max
- Temperatura di aspirazione: anomalia se il valore è fuori dal range previsto
- Temperatura ambiente: variabile esplicativa

Le anomalie consecutive relative alla stessa variabile vengono filtrate, mantenendo solo il valore più estremo per evitare duplicazioni inutili.

2.1 Invio dati all'assistenza tecnica

E' il vettore preferibile per una risposta umana e competente; ovviamente necessaria per gli interventi sul luogo.

Snippet di guasto fornito dal dispositivo:

[CONFIGURAZIONE IMPIANTO]

Setpoint: 2°C | Isteresi: 4°C Limiti allarme: -5°C / 10°C

[ESTRAZIONE DATI INLINE]

Ora 9: Condensatore 67.8°C SURRISCALDATO (limite: 45°C) | Termostato: 4.0°C | Evaporatore: -13.5°C | Condensatore: 67.8°C | Aspirazione: -9.6°C | Ambiente: 27.5°C | Porta aperta: no | Sbrinamento: no |



2.2 Invio dati all'AI

Ed è a questo punto che può intervenire l'AI per dare un contributo informativo all'utenza spesso disorientata da allarmi e codici di guasto esposti dal controllore e difficilmente interpretabili. L'elenco così consolidato e formattato come prompt efficace viene trasmesso all'AI insieme ai valori delle altre variabili, in modo da fornire gli elementi di correlazione necessari.

L'intelligenza artificiale non esegue alcun calcolo tecnico: si limita a produrre una risposta testuale, descrivendo il problema rilevato, le possibili cause e le conclusioni operative. In altre parole rende chiara la causa del problema quasi sempre affidata a codici poco comprensibili nei vari controllori presenti sul mercato. La stessa informazione può essere inviata al servizio di assistenza tecnica sia da Cloud che da telefono che ha ricevuto l'informazione tramite protocollo NFC finalizzata alla risoluzione del guasto da remoto, se possibile, o con intervento sul luogo.

In sintesi, il rilevamento delle anomalie è deterministico e basato su regole fisse, mentre l'AI interviene solo per l'interpretazione linguistica e la presentazione del risultato. Questo approccio completa il quadro della sorveglianza tecnica.

2.21 Richiesta API DeepSeek

Le richieste all'API DeepSeek seguono un formato standard che prevede l'utilizzo del metodo HTTP POST verso un endpoint dedicato. Ogni chiamata include l'header Content-Type: application/json e l'autorizzazione tramite Bearer Token. Il corpo della richiesta contiene la selezione del modello (deepseek-chat), i parametri di configurazione (max_tokens e temperature) e la sequenza di messaggi, in cui il contenuto del prompt viene inserito come messaggio utente.

```
POST https://api.deepseek.com/v1/chat/completions
```

Headers:

Content-Type: application/json

Authorization: Bearer

Body:

```
{  
  "model": "deepseek-chat",  
  "max_tokens": ,  
  "temperature": ,  
  "messages": [{"role": "user", "content": " " }]  
}
```

Obiettivo: eseguire la diagnosi di un frigorifero professionale TN ventilato a temperatura positiva, partendo da dati simulati.

Parametri API: max_tokens: 400, temperature: 0

2.22 Formato del Prompt

Il prompt è redatto in modo che l'AI assuma il ruolo di tecnico frigorista. Vengono fornite dettagliate informazioni sulla configurazione dell'impianto (tipologia, setpoint, isteresi, limiti di allarme, range normali, valori fuori limite). Le istruzioni specificano che la risposta deve basarsi esclusivamente su queste informazioni, senza introdurre dati non presenti, e fornire indicazioni sulle possibili cause di anomalie.

La risposta viene articolata in tre capitoli: ANALISI (riassunto dei parametri e anomalie), PROBLEMA RILEVATO (spiegazione tecnica della causa), CONCLUSIONI (consiglio operativo), con un massimo di 6 righe complessive.

2.23 Snippet di guasto filtrato

Ora 9: Condensatore 67.8°C SURRISCALDATO (limite: 45°C) | Termostato: 4.0°C | Evaporatore: -13.5°C | Condensatore: 67.8°C | Aspirazione: -9.6°C | Ambiente: 27.5°C | Porta aperta: no | Sbrinamento: no |

La risposta è un testo libero strutturato nelle sezioni ANALISI, PROBLEMA RILEVATO, CONCLUSIONI.

2.24 Risposta reale dell'AI

[ANALISI]

Anomalia rilevata: Condensatore a 67.8°C, surriscaldato oltre il limite normale (45°C). Tutti gli altri parametri (evaporatore, aspirazione) sono nei range.

[PROBLEMA RILEVATO]

Alta temperatura di condensazione. Possibili cause: sporcizia su condensatore, ventilatore condensatore guasto o difettoso, carica refrigerante eccessiva.

[CONCLUSIONI]

Intervento urgente: pulizia condensatore e verifica funzionamento ventilatore. Controllare eventuale sovraccarico di gas.

2.25 Costi AI

Costo Token (in+out) DeepSeek: 0.45 euro / [milione token]

Token scambiati: 753.

Costo in euro della singola transazione: 0,00033885 euro.

3. Sorveglianza complementare: Snapshot System

Lo Snapshot System è progettato come componente nativo di diagnostica professionale: raccoglie in un unico report lo stato dell'applicazione, della rete, del dispositivo e del controllore, con l'obiettivo di rendere immediata la comunicazione tra chi utilizza l'unità sul campo e chi sviluppa il software in azienda. L'utilizzatore tipico non è uno sviluppatore: è un professionista di altre discipline — cuoco, infermiere, pasticcere, farmacista o altre figure operative — che deve configurare l'unità, leggere i dati e segnalare un'anomalia in pochi minuti. Lo Snapshot System gli permette di farlo senza interpretare log o messaggi tecnici, generando automaticamente un report strutturato con contesto, modello, installazione, anomalie rilevate, richieste di rete recenti, errori JavaScript, informazioni sul dispositivo e log significativi, pronto per essere copiato e inviato all'assistenza con un solo gesto.

Nel nuovo scenario il controllore non comunica solo tramite NFC: può anche essere connesso al cloud. Questo significa che le informazioni raccolte dal termostato — utilizzo, interruzioni, stato dei relè, errori I2C, riavvii MCU, tensioni, temperatura PCB, release firmware — possono essere lette dall'utente tramite NFC oppure inviate automaticamente al cloud, dove diventano disponibili allo sviluppatore del software. L'azienda riceve una fotografia tecnica fedele del comportamento reale dell'unità, senza dipendere da interpretazioni soggettive.

Per lo sviluppatore, questo non rappresenta un "debug in produzione" nel senso tradizionale, perché il codice è già stato testato e validato prima della distribuzione. Tuttavia, il comportamento reale dipende da variabili che in laboratorio non sono completamente riproducibili: modelli di telefono diversi, browser differenti, versioni del sistema operativo, qualità della rete, condizioni ambientali, configurazioni specifiche dell'unità. Lo Snapshot System colma esattamente questo divario fornendo una diagnostica contestuale: una fotografia tecnica precisa del contesto in cui si manifesta un problema. Invece di ricevere segnalazioni generiche come "non funziona" o "si blocca", lo sviluppatore ottiene un report strutturato con errori JavaScript completi di stack trace, richieste HTTP con esito e tempi, stato della connessione, informazioni sul dispositivo, log recenti e, quando disponibile, i dati del controllore inviati dal Cloud.

Questo riduce drasticamente il tempo necessario per capire se l'anomalia è legata alla rete, a un modello specifico di telefono, a una configurazione particolare dell'unità, a un comportamento limite del firmware o a un problema hardware. Non si tratta di rifare il debug, ma di collegare rapidamente il problema al suo contesto reale, con dati oggettivi e immediatamente utilizzabili. L'integrazione NFC+Cloud permette inoltre di avere lo stesso snapshot sia sul campo sia in azienda, garantendo coerenza tra ciò che l'utente rileva e quello che lo sviluppatore analizza.



3.1 Funzioni dal modello di PWA IceSense

- generateSystemSnapshot: crea il report completo
- copySystemSnapshot: copia il report negli appunti
- switchAITab: attiva la generazione quando si apre il tab System.

La diagnostica si basa su tre meccanismi di intercettazione:

- interceptConsoleLogs per log e warning
- interceptJSErrors per errori JavaScript
- interceptNetworkRequests per le chiamate HTTP. trimInterceptedLogs mantiene i log entro un limite controllato

Le funzioni helper forniscono i dati necessari:

- getNetworkRequests per le richieste recenti
- getDeviceInfo e getDeviceInfoCached per il contesto hardware/software
- getPhoneModel e getDeviceInfoFromUserAgentData per identificare il dispositivo
- getOsDisplayString per la formattazione del sistema operativo

Le funzioni di ottimizzazione garantiscono fluidità e immediatezza:

- enrichDeviceInfoAtStartup prepara i dati del dispositivo
- generateSystemSnapshotInWorker elabora i log in background
- generateSystemSnapshotLazy genera il report solo al primo accesso

Lo Snapshot Functions utilizza due funzioni:

- generateFunctionsSnapshot per il report funzionale del controllore
- copyFunctionsSnapshot per la condivisione

Esempio di snapshot restituito dal controllore:

Acquisizione: 31/01/2026 17:00

Unità in esercizio: Via Severino Grattoni, 12 – Roma – Meteo locale 13°- 75%

Model: VITRA, S/n: A66566TY - Installato: 10/12/2025

[NETWORK]

17:00:22 GET <https://nominatim.openstreetmap.org/reverse> → 200

17:00:24 GET <https://api.open-meteo.com/v1/forecast> → 200

[JAVASCRIPT]

17:00:33 Test errore JavaScript:

at console.error (<https://iceeruda.pages.dev/:3842:14>) at toggleTestErrors
(<https://iceeruda.pages.dev/:4499:15>)

17:00:33 Test console.error:

errore di validazione dati at console.error (<https://iceeruda.pages.dev/:3842:14>) at toggleTestErrors
(<https://iceeruda.pages.dev/:4502:13>)

17:00:33 Unhandled Promise Rejection

[APP FLAGS]

API scaduta: Sì, Errore NFC scrittura: Sì

[DEVICE]

OPPO Reno 15F / Android 16 - Viewport: 1280×1024 | Network: 5G online

4. Sorveglianza complementare: Snapshot Functions

Lo Snapshot Functions è il sistema con cui il termostato registra e rende disponibili informazioni diagnostiche sull'unità refrigerante. Oggi mostra solo un modello statico; nel nuovo scenario raccoglie dati reali, li memorizza, li trasferisce all'app tramite NFC e, quando disponibile, li invia anche al Cloud. In questo modo il tecnico ottiene una diagnosi immediata sul campo, mentre l'ufficio hardware/firmware può analizzare gli stessi dati da remoto.

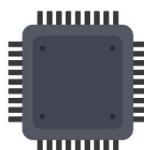
Il termostato registra: interruzioni e spegnimenti, stato dei relè, errori I2C, riavvii MCU, tensioni 3V3/12VDC, temperatura PCB, integrità firmware (checksum, watchdog, heap/stack, allocazioni fallite). Queste informazioni permettono al tecnico di valutare rapidamente l'affidabilità dell'unità, individuare componenti usurati, diagnosticare sensori guasti, riconoscere instabilità del firmware, verificare problemi di alimentazione o surriscaldamento e capire se intervenire subito o programmare una manutenzione. Il report può essere copiato e condiviso, evitando descrizioni verbali e permettendo confronti nel tempo.

Gli stessi dati, inviati al Cloud o condivisi dal tecnico, consentono all'ufficio hardware/firmware di analizzare pattern di utilizzo, valutare l'affidabilità dei relè, diagnosticare problemi sulle linee I2C, eseguire analisi post-mortem dei reset, verificare la stabilità dell'alimentazione in campo, valutare il design termico e identificare memory leak o buffer insufficienti. Questo permette debug remoto, identificazione di problemi ricorrenti, ottimizzazione firmware e validazione hardware in condizioni reali.

Il flusso dati è semplice: il termostato raccoglie e memorizza le metriche, le espone via NFC e, se connesso, le invia al Cloud. L'app legge i dati tramite NFC, li visualizza nella sezione Snapshot e permette la condivisione del report. Il Cloud li archivia e li rende disponibili all'ufficio tecnico per analisi e correlazioni.

Requisiti principali: memoria non volatile e aggiornamento continuo delle metriche lato termostato; formato dati standard (JSON/NDEF); lettura NFC e parsing lato app; endpoint di ricezione e archiviazione lato Cloud.

In sintesi, il tecnico ottiene una diagnostica immediata senza strumenti aggiuntivi; l'ufficio hardware/firmware riceve dati reali dal campo, può eseguire debug remoto e ottimizzare firmware e hardware; l'azienda beneficia di interventi più rapidi, maggiore qualità del prodotto e tracciabilità delle anomalie.



Esempio di snapshot restituito dal controllore:

Acquisizione: 31/01/2026 15:51

Unità in esercizio: Via Severino Grattoni, 12 – Roma

Meteo locale 13°- 75%

Model: VITRA, S/n: A66566TY

Installato: 10/12/2025

[UTILIZZO]

Interruzioni rete: 21

Spegnimenti: 5

Integrità mappe: si

[RELE]

Relè U1: score 70%

Relè U2: score 11%

Relè U3: score 54%

Relè U4: score 12%

score = [cicli effettuati / vita elettrica nominale (max commutazioni)] x 100

[ERRORE I2C]

Tipo ultimo errore: nack su indirizzo

Dispositivo coinvolto: 0x48

Reset bus I2C: 2

Errori totali sessione: 5

Stato linee in errore: SDA=LOW, SCL=HIGH

[RIAVVII MCU]

Tipo ultimo reset: watchdog

Registri di fault: 0xDEADBEEF

Stack pointer al crash: 0x3FFB7A10

Reboot totali: 3

[ALIMENTAZIONE 3V3]

Tensione attuale: 3280 mV

Tensione minima: 3190 mV

Tensione massima: 3310 mV

[ALIMENTAZIONE 12VDC]

Tensione attuale: 12040 mV

Tensione minima: 11890 mV

Tensione massima: 12110 mV

[MCU]

Temperatura PCB: 47 °C

[FIRMWARE]

Checksum firmware: OK (0xA3F2)

Watchdog: attivo, refresh OK

Stato task RTOS: 0x07

Heap libero: 18240 bytes

Stack libero: 4096 bytes

Allocazioni fallite: 0

Firmware release: 48901-56744

