



# GC-0006.01 – Sistema di acquisizione dati da sensori - Capitolato di Progetto

Authors: Moones Mobaraki, Cristian Pirlog, Luca Cossaro

Confidentiality: Public

Last Modified: 16/09/2025

Release Status: Released

M3I-UUID: NA



Codice/Titolo GC-0006.03 - Sistema di acquisizione dati da sensori -  
Capitolato di Progetto

Stato del Rilascio Released Classificazione Public

Data 17/09/2025 Pagina 2/14

## CONTROLLO DOCUMENTALE

<b>Titolo</b>	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto
<b>Classificazione</b>	Public
<b>Stato del Rilascio</b>	Released
<b>Data</b>	17/09/2025
<b>Riferimento M31</b>	Cristian Pirlog e-mail: cristian.pirlog@m31.com

## LISTA DI DISTRIBUZIONE

Società	Nome Cognome	Società	Nome Cognome
M31 S.r.l.			

## TABELLA DELLE REVISIONI

Rev	Data	Descrizione	Autori	Verificata	Approvata
01	11/09/2025	Prima revisione	Cristian Pirlog	Moones Mobaraki	Luca Cossaro
02	16/09/2025	Seconda revisione	Cristian Pirlog	Moones Mobaraki	Luca Cossaro
03	17/09/2025	Terza revisione	Luca Cossaro, Moones Mobaraki	Cristian Pirlog	Luca Cossaro
<b>Nome File:</b>	GC-0006.01 - Sistema di acquisizione dati da sensori - Capitolato di Progetto.docx				



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	3/14

## INDICE

<b>CONTROLLO DOCUMENTALE</b> .....	<b>2</b>
<b>LISTA DI DISTRIBUZIONE</b> .....	<b>2</b>
<b>TABELLA DELLE REVISIONI</b> .....	<b>2</b>
<b>INDICE</b> .....	<b>3</b>
<b>1 CONTESTO E OBIETTIVO DEL PROGETTO</b> .....	<b>4</b>
<b>2 DESCRIZIONE DEL SISTEMA</b> .....	<b>5</b>
<b>2.1 SENSORI BLE</b> .....	<b>5</b>
<b>2.2 GATEWAY BLE-WIFI</b> .....	<b>5</b>
<b>2.3 CLOUD</b> .....	<b>5</b>
<b>3 ARCHITETTURA</b> .....	<b>6</b>
<b>4 TECNOLOGIE DI RIFERIMENTO</b> .....	<b>8</b>
4.1 RIFERIMENTI.....	8
<b>5 REQUISITI</b> .....	<b>9</b>
5.1 REQUISITI FUNZIONALI MINIMI .....	9
5.2 REQUISITI FUNZIONALI OPZIONALI .....	9
5.3 REQUISITI DI SICUREZZA .....	9
5.4 REQUISITI DI SICUREZZA OPZIONALI .....	10
5.5 REQUISITI NON FUNZIONALI .....	10
<b>6 DELIVERABLE DEL PROGETTO</b> .....	<b>11</b>
6.1 ANALISI DELLO STATO DELL'ARTE .....	11
6.2 PROOF OF CONCEPT (POC).....	11
6.3 MINIMUM VIABLE PRODUCT (MVP).....	11
<b>7 TEST E VALIDAZIONE</b> .....	<b>12</b>
<b>8 DOCUMENTAZIONE</b> .....	<b>13</b>



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	4/14

## 1 CONTESTO E OBIETTIVO DEL PROGETTO

L'acquisizione e la gestione dei dati provenienti da sensori distribuiti è una sfida centrale in numerosi ambiti applicativi, come l'industria manifatturiera, la logistica, l'healthcare e le smart city. In particolare, l'utilizzo di sensori Bluetooth Low Energy (BLE) consente di raccogliere informazioni in modo capillare e a basso consumo energetico, abilitando scenari in cui decine o centinaia di dispositivi cooperano in tempo reale.

Tuttavia, la sola disponibilità dei dati non è sufficiente: è necessario disporre di una piattaforma in grado di ricevere, aggregare, normalizzare e smistare tali informazioni in modo affidabile, sicuro e scalabile. In assenza di un'infrastruttura strutturata, i dati rischiano di rimanere dispersi o non fruibili, riducendo fortemente il valore dell'intero ecosistema IoT.

Il progetto proposto mira quindi a sviluppare un sistema distribuito di acquisizione e smistamento dati da sensori BLE, articolato in tre livelli principali:

- **Sensori BLE:** dispositivi periferici che raccolgono dati dal campo (es. temperatura, umidità, movimento, segnali biometrici).
- **Gateway BLE-WiFi:** nodi intermedi che si connettono ai sensori, raccolgono dati tramite profili standard o personalizzati, li formattano secondo un modello interno e li inviano al cloud.
- **Cloud:** piattaforma centrale che gestisce la connessione sicura dei gateway, riceve e bufferizza i dati, li rende disponibili tramite API e interfacce di visualizzazione, garantendo segregazione tra diversi tenant.

Gli obiettivi principali del progetto sono:

- Progettare un'infrastruttura scalabile e sicura per la gestione di dati sensibili provenienti da sensori eterogenei.
- Garantire segregazione dei dati tra diversi tenant, consentendo a più soggetti di utilizzare lo stesso sistema in modalità multi-tenant.
- Implementare meccanismi di comunicazione sicura tra i vari livelli (gateway-cloud e cloud-client).
- Fornire strumenti di monitoraggio e visualizzazione dei dati sia per gli amministratori (gestione accessi, provisioning gateway) sia per gli utenti finali (dashboard e grafici in tempo reale).

Il progetto si presta a molteplici estensioni future, tra cui l'integrazione con sistemi di analisi avanzata e algoritmi predittivi, ma il focus principale sarà sulla raccolta, gestione sicura e distribuzione dei dati in uno scenario IoT complesso e realistico.



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	5/14

## 2 DESCRIZIONE DEL SISTEMA

### 2.1 Sensori BLE

- Dispositivi periferici già disponibili, non progettati nell'ambito del progetto.
- Devono essere interrogati tramite profili standard BLE (ad esempio GATT), con estensioni opzionali per protocolli custom.
- Producono dati grezzi da inviare al gateway.

### 2.2 Gateway BLE-WiFi

- Si connette ai sensori tramite profili BLE standard (obbligatorio) e, ove possibile, custom (opzionale).
- Raccolta dei dati in tempo reale dai sensori e conversione in un formato interno (da definire durante il progetto).
- Bufferizzazione dei dati in caso di perdita di connessione Internet (funzionalità opzionale).
- Invio dei dati al cloud tramite protocolli noti (es. HTTPS/MQTT con TLS).
- Supporto a meccanismi di provisioning sicuro (es. certificati digitali, chiavi di accesso).

### 2.3 Cloud

- Gestisce il provisioning sicuro dei gateway e la loro connessione autenticata.
- Implementa i protocolli di trasferimento dati dai gateway.
- Parsing e bufferizzazione dei dati in arrivo.
- Esposizione di API REST/GraphQL per la richiesta di dati già acquisiti.
- Esposizione di una streaming API per la ricezione in tempo reale dei dati.
- Gestione multi-tenant, garantendo segregazione tra dati di diversi clienti/utenti.
- Interfaccia web di gestione e configurazione (es. gestione gateway, utenti e permessi).
- Dashboard opzionale per la visualizzazione dei dati in tempo reale.
- Sistema di monitoraggio (es. Prometheus + Grafana) per valutare lo stato del sistema e raccogliere metriche.

I primi due livelli (sensori BLE e gateway) non sono oggetto di questo capitolato e devono essere considerati come strumenti esterni. Nel contesto del progetto, gli studenti non realizzeranno un gateway fisico ma svilupperanno un simulatore di gateway, in grado di riprodurre in maniera verosimile il comportamento di un nodo BLE-WiFi. Tale simulatore sarà utilizzato per validare l'infrastruttura cloud e per testare i flussi di comunicazione previsti.

L'oggetto del presente capitolato è il terzo livello: l'applicazione Cloud, tale applicazione dovrà essere sviluppata in accordo con i requisiti elencati nel capitolo 5.



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	6/14

### 3 ARCHITETTURA

L'architettura proposta è organizzata su tre livelli principali:

#### 3.1 Sensori BLE (Field Layer)

- Costituiscono i nodi periferici del sistema.
- Generano dati eterogenei (ad es. ambientali, industriali, biometrici).
- Sono dispositivi a basso consumo, spesso con capacità limitata di elaborazione.
- Comunicano esclusivamente tramite Bluetooth Low Energy, seguendo profili standard (GATT) o custom.

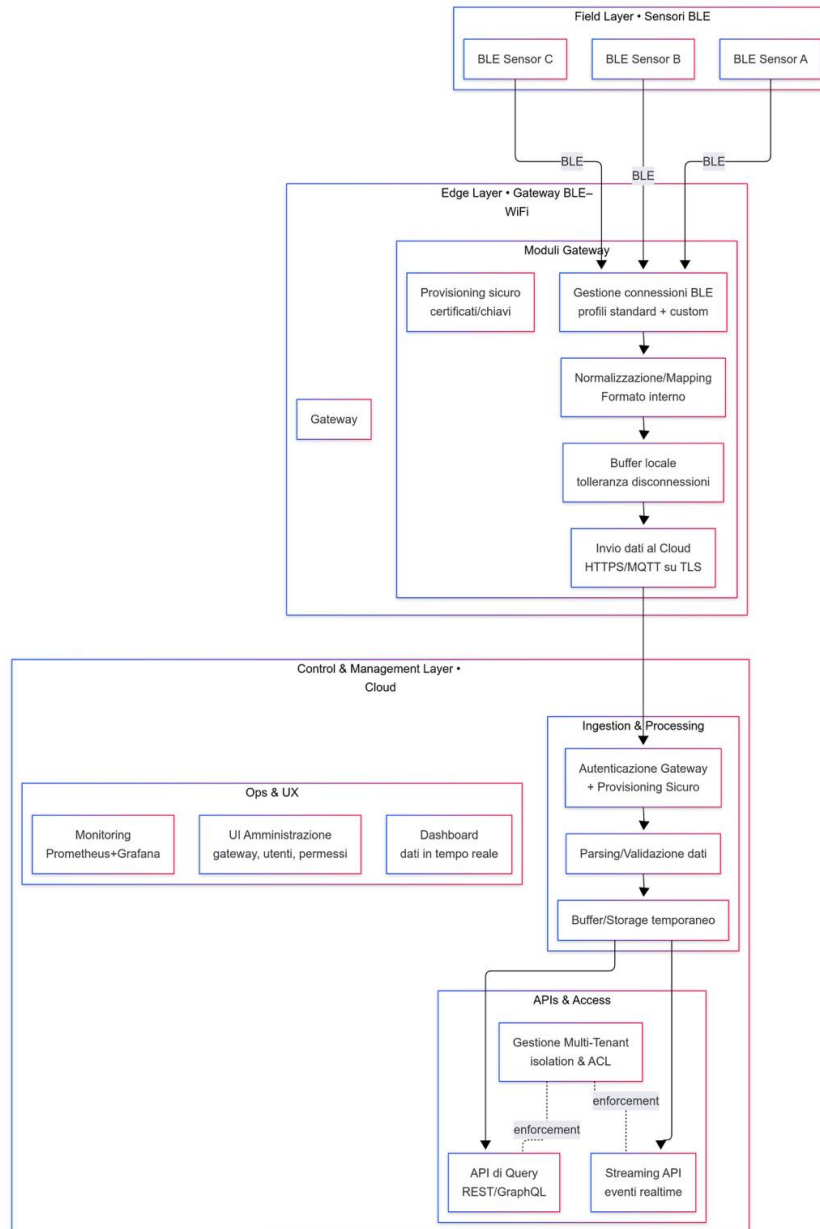
#### 3.2 Gateway BLE-WiFi (Edge Layer)

- Funzionano da punto di raccolta dei dati, con connessione diretta ai sensori tramite BLE.
- Ogni gateway elabora localmente i dati ricevuti, uniformandoli in un formato interno standardizzato, così da ridurre l'eterogeneità dei dispositivi sul campo.
- Devono supportare connessioni multiple a sensori, gestendo simultaneamente più stream di dati.
- Implementano buffer locale per tollerare disconnessioni temporanee dalla rete.
- Si collegano al cloud tramite WiFi (o altre tecnologie IP), utilizzando protocolli sicuri come **HTTPS** o **MQTT** su **TLS**.
- Devono essere provisionati in modo sicuro, associandosi a un tenant specifico e caricando certificati di autenticazione.

#### 3.3 Cloud (Control & Management Layer)

- È il cuore dell'infrastruttura e riceve i dati dai gateway autenticati.
- Garantisce la gestione multi-tenant, isolando i dati dei diversi clienti e applicando politiche di sicurezza e autorizzazione.
- Espone due tipologie di API:
  - API di query per richiedere dati storici o già bufferizzati.
  - API di streaming per ricevere i dati in tempo reale.
- Include un sistema di persistenza temporanea dei dati, con possibilità di estensione a database a lungo termine.
- Fornisce una console di amministrazione per la gestione dei gateway, degli utenti e delle autorizzazioni.
- Offre una dashboard opzionale per utenti finali, che consente di monitorare i dati raccolti e visualizzarli in tempo reale.
- È previsto l'uso di un monitoring system (Prometheus + Grafana) per osservare le prestazioni dell'infrastruttura e i volumi di traffico.

Tale architettura è da intendersi come indicativa e può essere soggetta a revisione o sostituzione con un'alternativa. In questi casi andrà motivato il cambiamento ed esso dovrà essere revisionato da M31.



**Figura 1 - Ipotesi di Architettura**



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	8/14

## 4 TECNOLOGIE DI RIFERIMENTO

Nel progetto proposto, sarà fondamentale utilizzare tecnologie scalabili e adatte alla gestione di sistemi distribuiti, in linea con quelle già adottate da M31. **Node.js** e **Nest.js** (usando **TypeScript** come linguaggio) verranno impiegati per lo sviluppo dei microservizi, grazie alla loro capacità di creare applicazioni server-side performanti e modulari. **Go** potrà essere utilizzato per eventuali componenti ad alte prestazioni, come i servizi di sincronizzazione. La comunicazione tra i microservizi sarà facilitata da **NATS** o **Apache Kafka**, entrambe tecnologie robuste per la gestione di messaggi distribuiti e asincroni.

Il sistema di orchestrazione e gestione centralizzata sarà ospitato su **Google Cloud Platform**, che offre servizi cloud affidabili e scalabili in particolare **Kubernetes**. Per l'archiviazione dei dati, si farà uso di **MongoDB** per la memorizzazione di dati non strutturati, mentre **PostgreSQL** potrà essere impiegato per la persistenza di dati strutturati. **Redis** potrà essere utilizzato come sistema di caching per migliorare le prestazioni e ridurre la latenza. Per quanto riguarda la parte di interfaccia utente si utilizza **Angular** e generalmente SPAs (Single Page Applications) per fornire agli utenti una esperienza il più possibile simile a quella di un'applicazione desktop.

Le tecnologie qui presentate non sono vincolanti, ma nel caso ne venissero proposte di alternative, queste scelte dovranno essere motivate ed esplicitamente accettate da M31. Lo stesso vale per eventuali aree tecnologiche non coperte da strumenti già in utilizzo presso M31.

### 4.1 Riferimenti

Tecnologie, linguaggi, strumenti e framework:

- NodeJS: <https://nodejs.org/>
- Nest.js: <https://nestjs.com/>
- TypeScript: <https://www.typescriptlang.org/>
- Go: <https://go.dev/>
- NATS: <https://nats.io/>
- Kafka: <https://kafka.apache.org/>
- Google Cloud Platform: <https://cloud.google.com/gcp>
- Kubernetes: <https://kubernetes.io/>
- PostgreSQL: <https://www.postgresql.org/>
- MongoDB: <https://www.mongodb.com/>
- Redis: <https://redis.io/>
- Angular: <https://angular.dev/>
- Architettura a Microservizi: <https://microservices.io/>



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	9/14

## 5 REQUISITI

### 5.1 Requisiti Funzionali Minimi

- RQ 1. Acquisizione e generazione dati da sensori:** il sistema deve poter gestire dati provenienti da sensori Bluetooth Low Energy (BLE). Poiché i sensori reali non sono oggetto del progetto, i dati saranno prodotti dal simulatore di gateway, che genererà valori realistici per 4-5 tipologie di sensori da concordare.
- RQ 2. Simulatore di gateway:** in sostituzione di un gateway fisico, deve essere sviluppato un simulatore in grado di riprodurre il comportamento principale:
- RQ 2.1.** generazione diretta dei dati simulati pronti alla trasmissione,
  - RQ 2.2.** invio al cloud tramite protocolli sicuri (SSL/TLS),
  - RQ 2.3.** gestione di più sensori simulati in parallelo,
  - RQ 2.4.** persistenza delle informazioni di commissioning (sensori/gateway) per garantire consistenza,
  - RQ 2.5.** capacità di rispondere ai messaggi dal cloud, anche in forma semplificata (con supporto opzionale a flag di debug che contengano la risposta attesa).
- RQ 3. Gestione multi-tenant:** il livello cloud deve garantire isolamento logico tra diversi tenant, permettendo a ciascun cliente/utente di accedere solo ai propri dati.
- RQ 4. Esposizione API centralizzate:** i dati raccolti dal simulatore e gestiti dal cloud devono essere accessibili tramite API sicure, documentate e versionate.
- RQ 4.1.** Ogni client deve essere identificato univocamente e certamente, questo per garantire il controllo dell'accesso ai dati e la loro segregazione
  - RQ 4.2.** Le API devono fornire accesso ai dati nelle seguenti modalità:
    - RQ 4.2.1.** On-Demand - accesso su richiesta a uno o più dati di un gateway, possibilmente con la possibilità di filtrare i dati per sensore e data/ora di ricezione.
    - RQ 4.2.2.** Stream - accesso ai dati in tempo reale attraverso uno stream continuo.
- RQ 5. Interfaccia utente:** deve essere fornita una UI web che consenta la consultazione e l'esplorazione dei dati acquisiti, con procedure semplificate per registrare e configurare nuovi sensori o gateway simulati.

### 5.2 Requisiti Funzionali Opzionali

- RQ 6. Integrazione con applicazioni di terze parti:** il sistema deve prevedere la possibilità di integrazione con applicazioni esterne tramite API o connettori dedicati, mediante l'utilizzo di API Key o Certificati SSL.
- RQ 7. Gestione eventi e notifiche:** devono essere previsti meccanismi di alert configurabili (via e-mail, dashboard, webhook) per notificare anomalie o eventi critici.

### 5.3 Requisiti di Sicurezza

- RQ 8. Autenticazione e autorizzazione:** tutti gli accessi al sistema devono essere autenticati tramite meccanismi robusti (JWT, OAuth2, mTLS) con ruoli granulari.



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	10/14

**RQ 9. Cifratura end-to-end:** i dati devono essere cifrati in transito e a riposo, utilizzando protocolli TLS e algoritmi standard di cifratura.

**RQ 10. Isolamento dei tenant:** i dati dei diversi tenant devono essere segregati sia a livello logico che fisico (database e storage dedicati o virtualizzati).

## 5.4 Requisiti di Sicurezza Opzionali

**RQ 11. Autenticazione a più fattori (MFA):** per l'accesso alla UI e alle API, deve essere prevista l'opzione di MFA.

**RQ 12. Audit e tracciamento:** il sistema deve mantenere log completi delle attività (accessi, modifiche ai dati, configurazioni), consultabili solo da utenti autorizzati.

**RQ 13. Resilienza agli attacchi:** devono essere implementati meccanismi di rate limiting, intrusion detection e failover automatico per garantire la continuità operativa.

## 5.5 Requisiti non Funzionali

**RQ 14. Scalabilità:** il sistema cloud deve progettato per essere scalabile orizzontalmente per gestire un numero crescente di sensori, gateway e tenant senza degrado delle prestazioni.

**RQ 15. Test automatizzati e code coverage:** lo sviluppo deve includere test unitari e di integrazione con un livello minimo di copertura da concordare ad inizio progetto.

**RQ 16. Monitoraggio:** devono essere predisposti strumenti di monitoraggio in tempo reale per le performance del sistema. In aggiunta, devono essere presenti almeno funzionalità di alert di base che consentano di capire rapidamente se un gateway sta funzionando correttamente o risulta non raggiungibile.

**RQ 17. Versionamento e DevOps:** il codice deve essere gestito tramite Git con pipeline CI/CD per il deploy su ambienti di test e produzione.



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	11/14

## 6 DELIVERABLE DEL PROGETTO

### 6.1 Analisi dello Stato dell'Arte

- Studio delle soluzioni IoT esistenti per l'acquisizione e la gestione di dati da sensori BLE, con particolare attenzione a piattaforme multi-tenant in cloud.
- Confronto tra diversi approcci architetturali (edge computing, serverless, microservizi) e tecnologici (NATS, Kafka, GCP Pub/Sub).
- Identificazione di best practices per la sicurezza dei dati IoT in scenari multi-tenant.

### 6.2 Proof of Concept (PoC)

- Realizzazione di un prototipo che dimostri la raccolta dati da sensori simulati tramite un **gateway simulato** e la trasmissione sicura verso il livello cloud.
- Dimostrazione delle funzionalità minime: cifratura dei dati, autenticazione dei dispositivi, sincronizzazione con un database.
- Per il PoC non è necessario realizzare prototipi fisici completi né un'infrastruttura cloud reale: è sufficiente utilizzare componenti simulati e un database locale, in modo da validare i concetti architetturali e i flussi principali senza l'onere di un'implementazione completa.

### 6.3 Minimum Viable Product (MVP)

- Implementazione di un sistema distribuito con più gateway e gestione multi-tenant nel cloud.
- Integrazione di un'interfaccia utente per la visualizzazione e l'esplorazione dei dati.
- Possibilità di gestire eventi, notifiche e conflitti di sincronizzazione.
- L'MVP utilizzerà esclusivamente il **simulatore di gateway** per la generazione dei dati e per la validazione delle funzionalità cloud. Non è richiesta la realizzazione di dispositivi fisici.



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	12/14

## 7 TEST E VALIDAZIONE

L'azienda M31 pone grande enfasi sulla qualità del codice e sulla validazione del sistema sviluppato attraverso test strutturati. Nel corso del progetto, dovrà essere prodotto un **test book** che descriva in dettaglio i test automatici implementati, i test manuali, i criteri di validazione e le metodologie utilizzate per garantire il corretto funzionamento del sistema in tutte le sue parti.

- **Test di integrazione sensore-gateway:** verificare la corretta acquisizione e trasmissione dei dati BLE.
- **Test di sincronizzazione cloud:** validare la consistenza dei dati tra i diversi gateway e il cloud.
- **Test di sicurezza:** includere penetration test e verifica dei meccanismi di cifratura/autenticazione.
- **Test di scalabilità e carico:** simulare un elevato numero di sensori/gateway per valutare la capacità del sistema di mantenere le prestazioni.
- **Test multi-tenant:** verificare l'isolamento e la segregazione dei dati tra tenant differenti.
- **Unit test e code coverage:** garantire un livello minimo concordato di copertura del codice.

Al termine del progetto i test descritti in questo test-book verranno eseguiti congiuntamente così da produrre un test-report finale. Anticipatamente all'esecuzione del test-book, M31 ed il committente si accorderanno su vincoli necessari a considerare il test-report come valido ed accettabile, in particolare verrà definita la quantità minima di test con risultato PASS che sarà necessaria per M31 per accettare quanto sviluppato.



Codice/Titolo	GC-0006.03 - Sistema di acquisizione dati da sensori - Capitolato di Progetto		
Stato del Rilascio	Released	Classificazione	Public
Data	17/09/2025	Pagina	13/14

## 8 DOCUMENTAZIONE

Per tutto il ciclo di vita del progetto sarà necessario produrre una documentazione dettagliata che comprenda:

- **Diagrammi architetturali:** sarà richiesta la produzione di diagrammi a blocchi per descrivere l'architettura del sistema e i vari componenti coinvolti (Field Layer, Edge Layer, Cloud Layer, tenant e applicazioni). Si consiglia di utilizzare tool quali C4 (<https://c4model.com/>) per la descrizione dell'architettura nei suoi vari livelli e Mermaid (<https://mermaid.js.org/>) per grafici più specifici o di processo.
- **Documentazione tecnica:** la documentazione del codice e delle funzionalità sviluppate dovrà essere completa, chiara e facilmente consultabile. Includerà descrizioni delle API esposte, delle procedure di deploy su Kubernetes, delle configurazioni di autenticazione e sicurezza, e delle pipeline CI/CD utilizzate.
- **Manuale dei test:** il test book dovrà essere redatto in modo esaustivo e descrivere tutte le procedure di test adottate, i criteri di successo e i risultati ottenuti durante la fase di validazione.
- **Manuale utente e amministratore:** dovrà essere prodotta una guida pratica che illustri le modalità di utilizzo dell'interfaccia utente per l'esplorazione dei dati, le API disponibili per le integrazioni esterne e le procedure di gestione dei tenant da parte degli amministratori di sistema

Copyright M31 © 2023 - This material is protected by copyright controlled by M31 S.r.l..

All rights are reserved. Unless otherwise stated, all product names, logos, and brands are property of their respective owners. All company, product and service names used in this material are for identification purposes only. Use of these names, logos, and brands does not imply endorsement. Copying, including reproducing, storing, adapting or translating, any or all of this material is strictly forbidden and requires the prior written consent of M31 S.r.l.. This material may also contain confidential information, which may not be disclosed to others without the prior written consent of M31 S.r.l..

M31 S.r.l.  
via Tommaseo, 77  
35131 Padova (PD) - Italy  
Tel: +39 049 78 11 811  
Fax: +39 049 78 11 899



[info@m31.com](mailto:info@m31.com) - [www.m31.com](http://www.m31.com)