Sommario

[GPSReader Project Documentation 2](#_Toc192656928)

[1. Requisiti Funzionali 2](#_Toc192656929)

[2. Architettura 2](#_Toc192656930)

[**2.1 Modulo di Parsing NMEA** 2](#_Toc192656931)

[**2.2 Modulo di Acquisizione Dati** 3](#_Toc192656932)

[**2.3 Interfaccia Utente** 3](#_Toc192656933)

[**2.4 Altri Moduli** 3](#_Toc192656934)

[3. Diagrammi UML 4](#_Toc192656935)

[3.1 Diagramma delle Classi (PlantUML) 4](#_Toc192656936)

[3.2 Diagramma dei Componenti 5](#_Toc192656937)

[3.3 Diagramma di Sequenza 6](#_Toc192656938)

**GPSReader Project Documentation**

**1. Requisiti Funzionali**

Il progetto è un’applicazione per la ricezione, il parsing e la visualizzazione dei dati GPS tramite il protocollo NMEA 0183. I requisiti funzionali principali includono:

* **Ricezione dei Dati GPS:**
  + Acquisizione dei dati provenienti da una porta seriale (o altre interfacce come FIFO, socket) in formato NMEA 0183.
  + Gestione di modalità operative diverse (standalone e plugin).
* **Parsing dei Messaggi NMEA:**
  + Interpretazione di tutte le frasi NMEA principali, tra cui:
    - **GPRMC, GPGGA, GPVTG, GPGLL, GPGSA, GPGSV, GPTXT, GPZDA, HDG, HDM, HDT** e altri messaggi proprietari o standard.
  + Estrazione di tutti i campi per ciascuna frase secondo lo standard NMEA 0183 (con nomi descrittivi, ad esempio "utcTime", "latitude", "fixQuality", ecc.).
  + Utilizzo di un framework modulare e templatizzato che consenta di aggiungere facilmente nuovi parser.
* **Visualizzazione dei Dati:**
  + Mostrare in tempo reale i dati grezzi ricevuti.
  + Permettere, tramite l’interfaccia grafica, di visualizzare il dettaglio completo di ogni frase (tutti i campi, con i relativi nomi e valori) quando l’utente seleziona una riga nel log.
  + Visualizzazione delle coordinate elaborate (in gradi decimali) e dei messaggi di stato.
* **Controllo e Configurazione:**
  + Possibilità di mettere in pausa, riprendere o riavviare la ricezione dei dati.
  + Caricamento dei parametri di configurazione (modalità, porta, baud rate, ecc.) da un database.
* **Supporto per Interfacce Multiple:**
  + Implementazione di canali di comunicazione alternativi (FIFO, socket) per modalità plugin.

**2. Architettura**

L’architettura del sistema è progettata in maniera modulare per garantire flessibilità ed estendibilità. I principali moduli sono:

**2.1 Modulo di Parsing NMEA**

* **Interfaccia Base e Classe Generica:**
  + **INMEASentenceParser:** Interfaccia che definisce i metodi essenziali per ogni parser.
  + **GenericNMEASentenceParser:** Classe template che implementa il pre‑parsing comune (verifica iniziale, rimozione del checksum, tokenizzazione) e delega il parsing specifico al metodo parseTokens della classe derivata.
* **Parser Specifici:**
  + Ogni frase principale (es. GPRMC, GPGGA, GPVTG, GPGLL, GPGSA, GPGSV, GPTXT, GPZDA, HDG, HDM, HDT) ha il suo parser che estrae i campi secondo lo standard NMEA 0183 e li memorizza in una mappa (QMap) con chiavi descrittive.
  + Per le frasi non implementate è previsto un **GenericFallbackParser** che memorizza ogni token con chiavi “field0”, “field1”, ecc.
* **Facade – NMEAParser:**
  + Classe statica che, dato un raw sentence, identifica il tipo e crea dinamicamente il parser specifico (o il fallback) per restituire un oggetto ParsedData contenente:
    - Il tipo di frase
    - La mappa dei campi
    - Un flag di validità
    - Un eventuale messaggio di errore

**2.2 Modulo di Acquisizione Dati**

* **GPSReader:**
  + Si occupa di leggere i dati dalla porta seriale (o altre interfacce) e di suddividerli in righe.
  + Per ogni frase NMEA ricevuta, utilizza il modulo di parsing (NMEAParser) per interpretare il messaggio.
  + Emissione di segnali:
    - rawDataReceived (dati grezzi)
    - parsedDataReceived (dati parsati)
    - newCoordinate (coordinate elaborate per GPRMC)
    - statusUpdated (messaggi di stato)

**2.3 Interfaccia Utente**

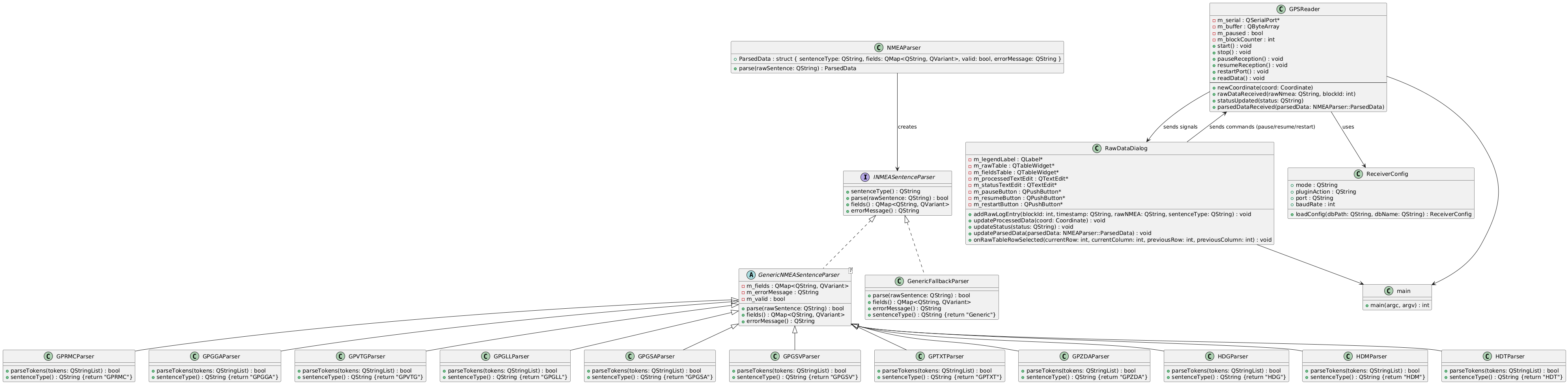
* **RawDataDialog:**
  + Mostra il log delle frasi NMEA ricevute con informazioni come timestamp, raw sentence e tipo di frase.
  + Quando l’utente seleziona una riga, la GUI esegue il parsing (se non già memorizzato) e mostra in una tabella il dettaglio completo (campi e relativi valori).
  + Fornisce controlli (pulsanti) per mettere in pausa, riprendere o riavviare la ricezione.

**2.4 Altri Moduli**

* **ReceiverConfig:**  
  Gestisce il caricamento della configurazione dal database.
* **Componenti di Comunicazione Alternativa:**  
  Come FIFODataSource, FIFOWriter e SocketChannel, per modalità plugin.

# 3. Diagrammi UML

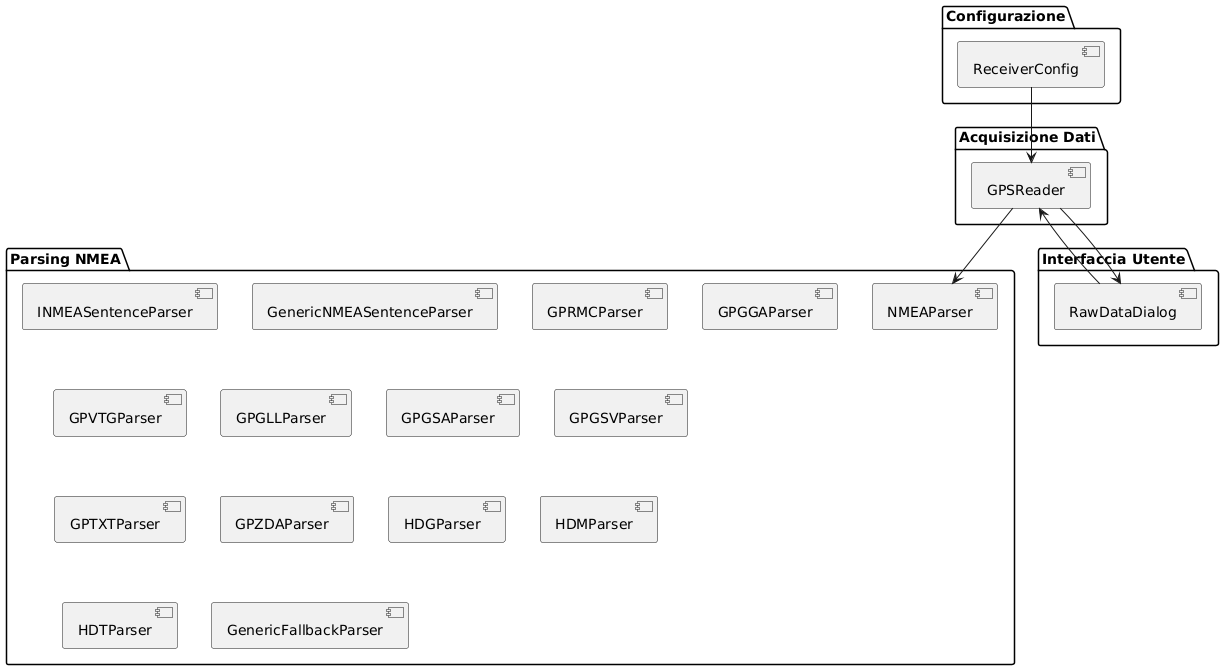
## 3.1 Diagramma delle Classi (PlantUML)



## 3.2 Diagramma dei Componenti

Il sistema può essere suddiviso nei seguenti componenti:

* **Modulo di Acquisizione (GPSReader):**  
  Gestisce la comunicazione seriale e l’emissione dei segnali di dati grezzi e parsati.
* **Modulo di Parsing (Parser Framework):**  
  Comprende le classi di parsing (INMEASentenceParser, GenericNMEASentenceParser, parser specifici, fallback e NMEAParser) che trasformano il raw NMEA in dati strutturati.
* **Modulo di Interfaccia Utente (RawDataDialog):**  
  Visualizza i log delle frasi e i dettagli dei campi, e fornisce controlli per la gestione della ricezione.
* **Modulo di Configurazione (ReceiverConfig):**  
  Carica la configurazione da un database.
* **Componenti di Comunicazione Alternativa (opzionali):**  
  FIFODataSource, FIFOWriter, SocketChannel per modalità plugin.



## 3.3 Diagramma di Sequenza

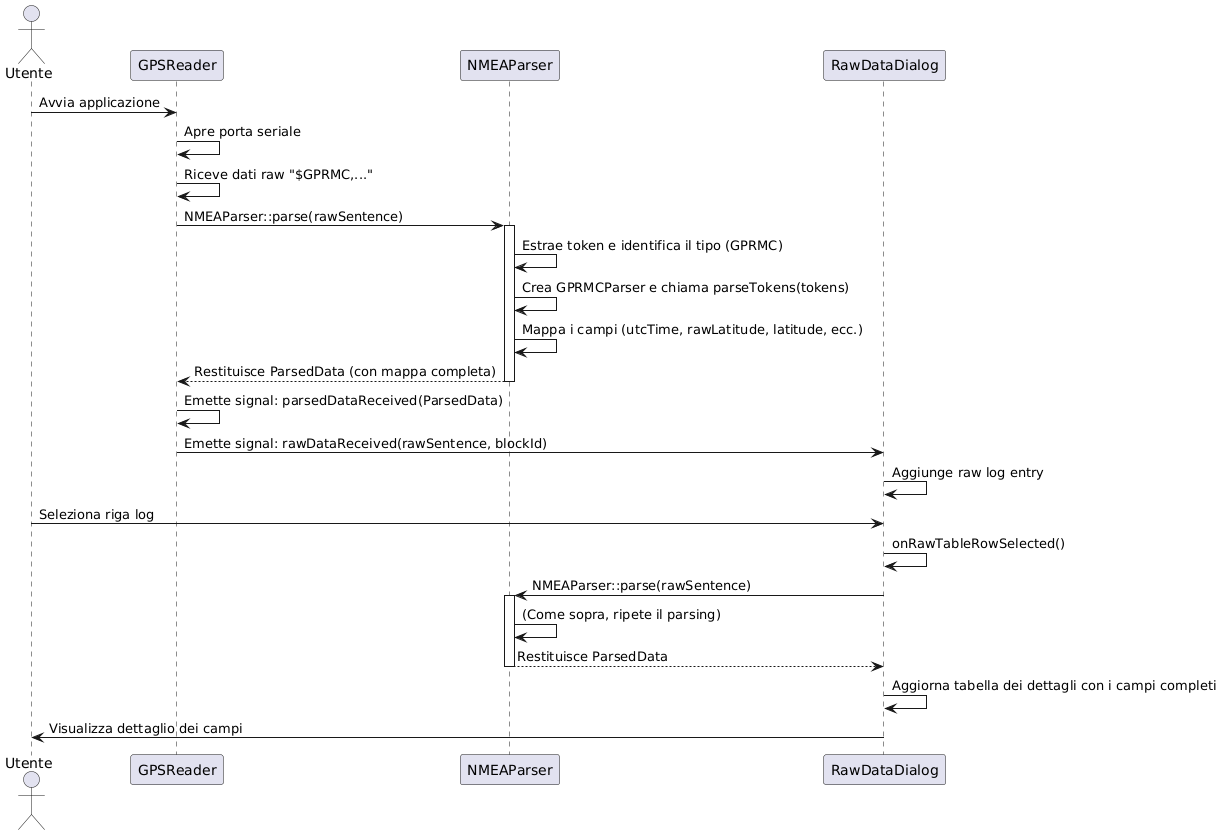
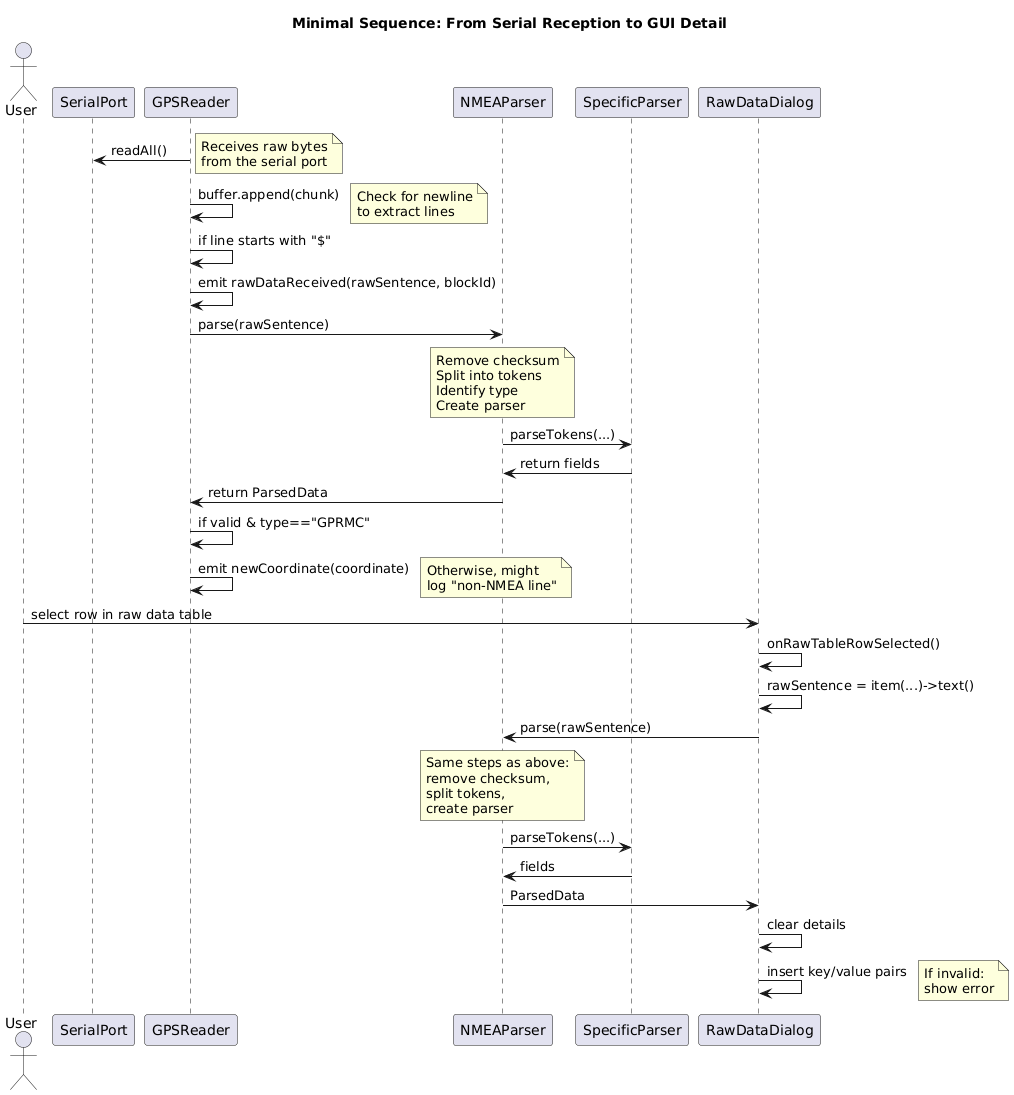
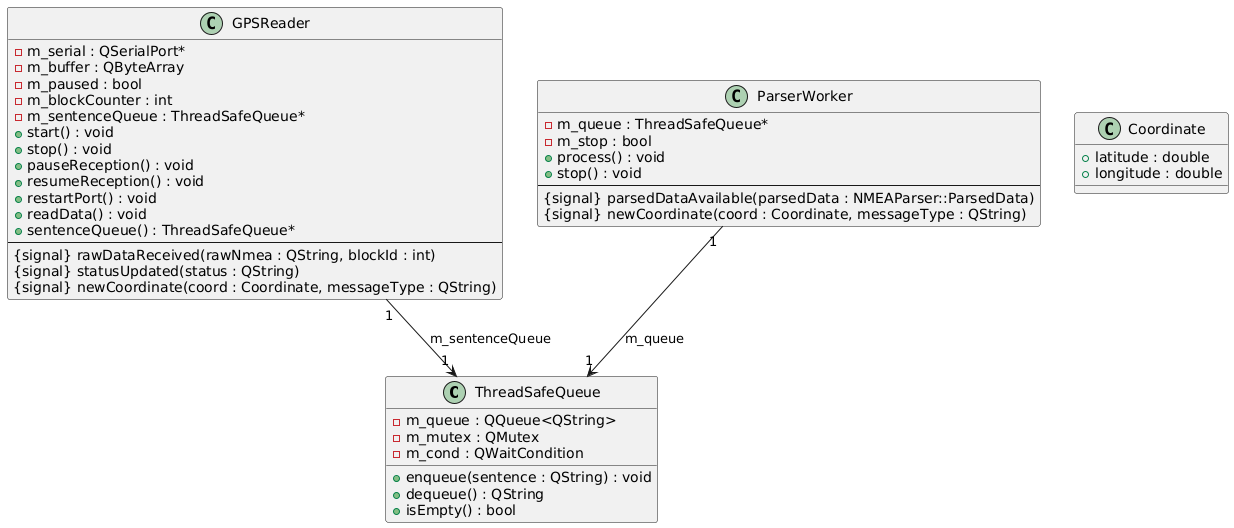


Diagramma 2



Questo diagramma mostra le classi principali del modello Producer–Consumer: la coda thread-safe, il Producer (GPSReader), il Consumer (ParserWorker) e la struttura Coordinate.



Questo diagramma di sequenza mostra il flusso completo: dalla lettura dei dati sulla seriale (GPSReader), alla loro inserzione nella coda (ThreadSafeQueue), al consumo in un thread separato (ParserWorker) e all'aggiornamento della GUI (RawDataDialog).

