

RELAZIONE – HOMEWORK 2 - Prova in Itinere DSBD aa 2024-2025

Studenti:

Bonafede Salvatore Luca (1000067612)

Bontempo Gaetano (1000067613)

ABSTRACT

Per l'Homework 2, il sistema sviluppato è stato esteso con nuove funzionalità per migliorare la gestione dei dati e introdurre un meccanismo di notifiche (tramite e-mail).

L'obiettivo principale è stato l'adozione del pattern **CQRS** (Command Query Responsibility Segregation), che separa la gestione delle operazioni di lettura e scrittura sul database, è stato implementato creando due classi **CommandHandler** e **Queryhandler**, il primo si occupa delle operazioni di scrittura mentre il secondo di lettura. Inoltre, il sistema è stato migliorato attraverso l'uso di Kafka come broker di messaggi, garantendo maggiore scalabilità.

Durante la registrazione, gli utenti possono ora fornire due nuovi parametri: **high-value** e **low-value**, che rappresentano rispettivamente la soglia superiore e inferiore per i valori dei ticker finanziari di loro interesse. Tali parametri possono essere successivamente aggiornati tramite operazioni gRPC. Il database è stato modificato per memorizzare questi valori, con la condizione che **high-value** debba essere maggiore di **low-value** se entrambi i parametri sono forniti.

Il **Data Collector** è stato aggiornato con l'obiettivo di non solo recuperare e aggiornare i valori dei ticker registrati nel database, ma anche inviare un messaggio su un topic Kafka (*to-alert-system*) per notificare che l'aggiornamento dei dati è stato completato.

(N.B. il data collector invia richieste a yfinance solamente quando il mercato è aperto)

Sono stati implementati due nuovi componenti aggiuntivi:

- **AlertSystem**: Questo componente consuma i messaggi dal topic Kafka *to-alert-system*. Una volta ricevuto il messaggio di completamento dell'aggiornamento del database, esegue una scansione per verificare se i valori dei ticker superano le soglie impostate dagli utenti. Se una delle soglie (alta o bassa) è superata, *AlertSystem* invia un messaggio al topic Kafka *to-notifier*, contenente le informazioni relative alla condizione di superamento della soglia.

- **AlertNotifierSystem:** Questo componente consuma i messaggi dal topic Kafka *to-notifier*. Dopo aver ricevuto il messaggio, invia un'email all'utente, notificandogli che il valore del ticker ha superato una delle soglie. L'email specifica se il valore ha superato la soglia alta (*high-value*) o quella bassa (*low-value*), permettendo così all'utente di essere avvisato tempestivamente del cambiamento.