

Sistemi e Applicazioni Cloud – Laboratorio

Appello 05 novembre 2025

Tempo consegna: 2h 30m

Si realizzi un'applicazione su Google Cloud Platform per la gestione delle prenotazioni dei pasti di un servizio mensa destinato agli asili. Gli educatori di ciascun asilo devono poter inserire, per una certa data, il numero di bambini che consumeranno il pasto, indicando il nome dell'asilo. Per la stessa data devono poter essere effettuate correzioni da parte di un asilo; il sistema deve essere in grado di sovrascrivere l'entry precedentemente inserita nel caso di una nuova entry dallo stesso asilo con la stessa data.

Il sistema mensa deve poter consultare, per una determinata giornata, il numero totale di pasti da preparare e il dettaglio degli asili ordinanti.

Si rendano disponibili i seguenti tipi di interfacce di utilizzo:

1. API per inserire e correggere le prenotazioni di pasto;
2. Pagina Web per la visualizzazione degli ordini giornalieri da parte della mensa.

L'applicazione deve essere testata per il deployment su piattaforma GCP utilizzando i servizi visti a lezione:

- App Engine
- Firestore
- Functions

Parte 1 – Backend API REST

L'applicazione deve esporre le seguenti funzionalità tramite opportune Web API RESTful:

1. Inviando richieste POST all'URI `/api/v1/mensa/{data}` è possibile registrare una prenotazione di pasti per un asilo nido in una determinata data. Il corpo della richiesta deve contenere almeno:
 - **asilo**: nome stringa dell'asilo nido (campo obbligatorio);
 - **bambini**: numero di bambini/pasti richiesti per quella data (intero positivo).

Se per lo stesso asilo e la stessa data esiste una prenotazione, l'invio di una nuova richiesta comporta un errore di duplicazione.

2. Inviando richieste PUT all'URI `/api/v1/mensa/{data}` è possibile aggiornare una prenotazione di pasti per un asilo nido in una determinata data. Il corpo della richiesta deve contenere almeno:

- **asilo**: nome stringa dell'asilo nido (campo obbligatorio);
- **bambini**: numero di bambini/pasti richiesti per quella data (intero positivo).

Il numero inserito sovrascrive quello precedente per lo stesso asilo e la stessa data. Se per lo stesso asilo e la stessa data non esiste una prenotazione, l'invio della richiesta deve generare un errore.

3. Inviando richieste GET allo URI `/api/v1/mensa/{data}` il sistema mensa può ottenere:

- il totale dei pasti da preparare per quella data;
- la lista degli asili che hanno effettuato la prenotazione e il numero di pasti confermati per ciascuno (una sola entry per asilo).

4. inviando richieste POST allo URI `/api/v1/clean` è possibile cancellare gli ordini memorizzati nel database

L'interfaccia di utilizzo delle API deve rispettare rigorosamente il file di specifica *OpenAPI* disponibile insieme alla presente specifica.

Per testare la funzionalità dell'API è possibile far riferimento al seguente URL:

<https://sacvalidator2025.appspot.com/>

Parte 2 – Web Application

Realizzare due pagine Web:

1. Realizzare due pagine web destinate ai gestori del servizio mensa:

- (a) Una pagina con il riepilogo dei pasti preparati nell'ultima settimana con in evidenza i pasti previsti per la giornata odierna
- (b) Una pagina con il dettaglio del singolo ordine

La pagina con la lista dei pasti della settimana deve includere un link per accedere ai dettagli della data specifica, compresa la data odierna in evidenza.

2. La pagina di dettaglio di una giornata deve contenere:

- La lista degli asili che hanno effettuato un ordine con il numero di pasti ordinati
- Numero di pasti richiesti nel giorno

Parte 3 – Interconnessione di servizi

Realizzare un sistema per mantenere aggiornato automaticamente il totale giornaliero dei pasti.

In particolare si vuole usare una funzione che:

- abbia come *trigger* l'inserimento o l'aggiornamento di una prenotazione nel database (relativo a una coppia *(data, asilo)*);
- abbia come azione l'aggiornamento di un documento di riepilogo giornaliero che contiene:
 - la data;
 - il totale dei pasti della giornata;
 - lista degli ordini da parte degli asili.
- creare una **web function** che risponda all'url `/trend/{data}` e restituisca il trend della data in questione rispetto alla media dei 7 giorni precedenti (vedi sezione "Definizione di trend").
- inviando una richiesta GET all'URI `/api/v1/trend/{data}` che restituisca il trend della data specificata rispetto alla settimana precedente.

Non è necessario collegare direttamente la funzione all'applicazione Web: è sufficiente mostrare che il database dei dati di riepilogo si popoli e si aggiorni correttamente quando cambiano le prenotazioni.

Note

- Se la validazione dell'API con il validatore non è positiva, l'esame è automaticamente insufficiente.
- In caso di validazione positiva, il voto sarà basato su quante parti dell'esame sono state portate a termine e sulla qualità del codice e delle soluzioni proposte.

Dettagli implementativi

Per l'implementazione si ricorda che:

- per memorizzare le prenotazioni dei pasti si può usare il database Firestore con una collezione **prenotazioni** in cui ciascun documento ha come identificatore una combinazione della data e del nome dell'asilo (ad es. `2025-11-05_asilo-arcobaleno`);
- per memorizzare i riepiloghi giornalieri si può usare una collezione **riepiloghi** in cui ciascun documento ha come nome la data del pasto e contiene il totale dei pasti, la lista degli asili con i relativi numeri e il valore di trend;
- si consiglia di sviluppare una funzione *helper* che, dato `(data)`, legga dal database i valori della settimana precedente alla data corrente (meno -7 giorni è considerato corretto) e calcoli la media delle richieste per calcolare quindi il valore del trend `((trend))`;

- per risparmiare tempo, si suggerisce di implementare sia le API che l'applicazione Web in un'unica sorgente e di farne il deployment come unico servizio di default;
- le date sono in formato **giorno-mese-anno** per l'interfaccia utente. Per le conversioni si possono usare le funzioni elencate di seguito.

Funzioni per gestire le date

```
from datetime import datetime

def date_from_str(d):
    # converte 'gg-mm-YYYY' in oggetto datetime
    try:
        return datetime.strptime(d, '%d-%m-%Y')
    except:
        return None

def str_from_date(d):
    # converte oggetto datetime in 'gg-mm-YYYY'
    return d.strftime('%d-%m-%Y')
```

Definizione di trend

Il **trend di un numero** rappresenta la variazione di un valore rispetto a un periodo di riferimento, utile a capire se sta crescendo, diminuendo o rimanendo stabile. Nel caso degli **ordini giornalieri**, possiamo confrontare il numero di ordini di oggi con la media della settimana precedente.

La variazione percentuale si calcola come:

$$\text{Trend_\%} = \frac{\text{Oggi} - \text{Media_settimana_prec}}{\text{Media_settimana_prec}} \times 100$$

Questo valore mostra di quanto, in percentuale, gli ordini siano aumentati o diminuiti rispetto alla settimana precedente. Se nella settimana precedente non sono presenti ordini, il trend restituisce 0.