

Prova d'esame di Laboratorio - Sistemi e Applicazioni Cloud

Appello 8 giugno 2020 [Tempo consegna: 2h30]

Si realizzi un'applicazione web su Google Cloud Platform per la gestione di uno smart garden. L'applicazione permette agli utenti di monitorare la semina delle piante e di mantenere delle informazioni delle piante, permettendo inoltre di gestire l'irrigazione automatica del giardino.

Si rendano disponibili i seguenti tipi di interfacce di utilizzo:

1. API per l'inserimento e la visualizzazione delle informazioni piante e delle informazioni di semina;
2. Interfaccia HTML di visualizzazione del giardino.

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

- App Engine
- Firestore
- Endpoints
- PubSub
- Functions

API REST [Sbarramento]

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

1. inviando richieste **POST** allo URI `/api/v1/garden/plant/{date}/{plant}` è possibile inserire nel programma la semina di una nuova pianta;
2. inviando richieste **GET** allo URI specificato in precedenza è possibile ottenere le informazioni riguardanti la semina indicata.

L'interfaccia di utilizzo delle API deve soddisfare **rigorosamente** il file di specifica *OpenAPI* disponibile sul sito del corso¹.

Progettazione [Voto: C]

Definire le specifiche *OpenAPI* per l'implementazione di un'ulteriore API che consenta di ottenere le informazioni di dettaglio della pianta. La nuova API dovrà essere quanto più integrata con quelle esistenti ed esporre un'interfaccia simile. Si richiede inoltre di implementare l'API secondo le specifiche definite.

Web Application [Voto: B]

Realizzare un'applicazione web che permetta di visualizzare i dettagli temporali delle semine secondo i dati inseriti precedentemente. La pagina dovrà presentare le informazioni temporali di semina, prevista germinazione, e prevista crescita completa in maniera simile alla modalità visualizzazione **programmazione** del Calendario Google, evidenziando in grigio le date passate, in azzurro le date prossime (10 giorni a partire da oggi), e in blu le future date.

¹https://weblab.ing.unimore.it/sac/1920/lab/esami/garden_openapi.yaml

Interconnessione di servizi [Voto: A]

Realizzare un sistema per la gestione automatizzata dell'irrigazione tramite l'utilizzo di un'interfaccia di rete compatibile con il servizio *Google Cloud PubSub*. Tale servizio funziona in duplice modalità:

1. Attivazione manualmente tramite invio di un messaggio contenente informazioni sulla durata dell'irrigazione (espressa in secondi) su un topic specifico;
2. Attivazione automatica tramite analisi dell'umidità del terreno, attivando l'irrigazione quando l'umidità scende sotto il 30%. Il valore di umidità viene letto su un topic diverso dal precedente.

Simulare il comportamento dell'irrigatore tramite *Cloud Functions*, stampando messaggi di informazione riguardo lo stato dell'irrigatore:

- acceso manuale;
- acceso automatico;
- spento.

Extra [Voto: S]

Realizzare uno script Python locale per simulare il sensore di umidità. Tale script dovrà interfacciarsi agli stessi topic *PubSub* utilizzati in precedenza e inviare il valore di umidità del suolo secondo le seguenti regole:

1. In assenza di irrigazione, l'umidità del terreno cala del 5% ogni secondo;
2. Durante l'irrigazione manuale, l'umidità del terreno aumenta del 10% ogni secondo;
3. Al termine dell'irrigazione automatica l'umidità del terreno deve essere del 100%.

Note

- Il mancato soddisfacimento **rigoroso** dei requisiti specificati nel file *OpenAPI* comporta l'invalidazione diretta della prova di laboratorio.
- A tale scopo, il soddisfacimento dei requisiti è testato con un correttore automatico disponibile all'indirizzo <https://sac-backend-validation.appspot.com>.