

Appello 15 Maggio 2017 - Soluzione Completa



Testo dell'Esame

Una giovane startup italiana vuole creare un **servizio web** che consenta ad un qualsiasi utente di **salvare una stringa associandola ad una chiave**. Il salvataggio avviene nel cloud della startup.

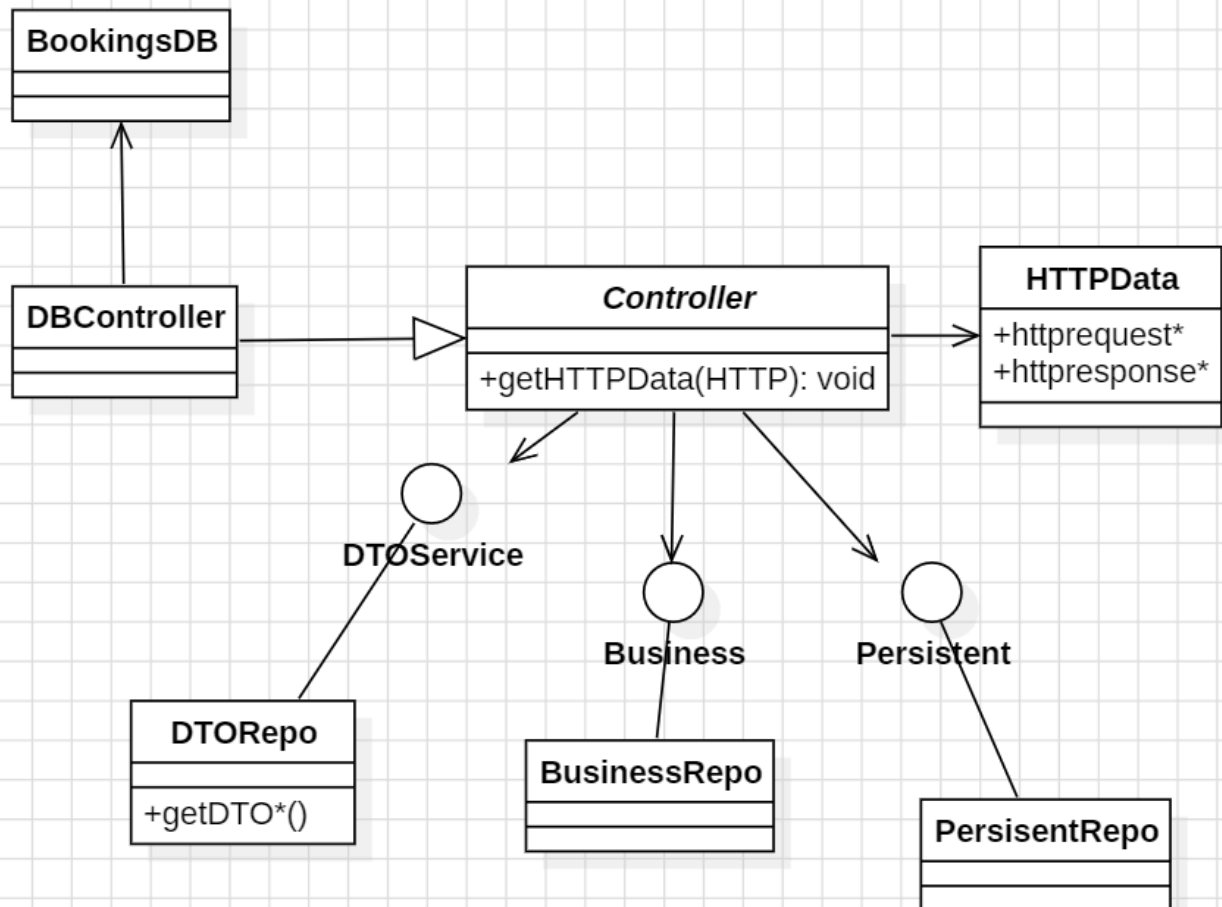
Il servizio deve esporre un'**interfaccia REST** che attraverso l'utilizzo di richieste HTTP con i verbi **HTTP POST e GET** permettano rispettivamente di **salvare e recuperare** successivamente la coppia (key, value).

La **struttura dati fornita** con le richieste (HTTP body) è simile al seguente **JSON** `{'key': 'chiave', 'value': 'valore'}`. Esso viene **processato per creare una struttura dati opportuna** che ne consente una più semplice elaborazione.

Il servizio REST è implementato utilizzando un **listener** che resta in ascolto di connessioni sulla **porta 80** nel caso di connessioni in HTTP. Esiste inoltre un'estensione del listener che utilizza la **crittazione (HTTPS)**, restando in ascolto sulla **porta 443**.



Soluzione



Controller → gestisce HTTP e coordina tre servizi distinti:

1. **DTOService + DTORepo:**

- Si occupa della trasformazione dei dati HTTP in DTO
- La DTORepo probabilmente contiene il metodo `getDTO()` per convertire i dati HTTP in oggetti DTO

2. **Business + BusinessRepo:**

- Gestisce la logica di business dell'applicazione
- BusinessRepo potrebbe contenere le regole di validazione e elaborazione

3. **Persistent + PersistentRepo:**

- Si occupa della persistenza dei dati nel database
- È qui che avviene il salvataggio effettivo e dove si genera l'ID per l'HTTP 201

Il flusso per HTTP 201 Created sarebbe:

```

HTTPData (request) → Controller →
├─ DTOService → DTORepo (conversione JSON→DTO)
├─ Business → BusinessRepo (validazione/elaborazione)
└─ Persistent → PersistentRepo (salvataggio → ID generato)
  
```

Poi la risposta risale:

```
PersistentRepo (ID) → Persistent → Controller → HTTPData (response 201 +  
Location header)
```