

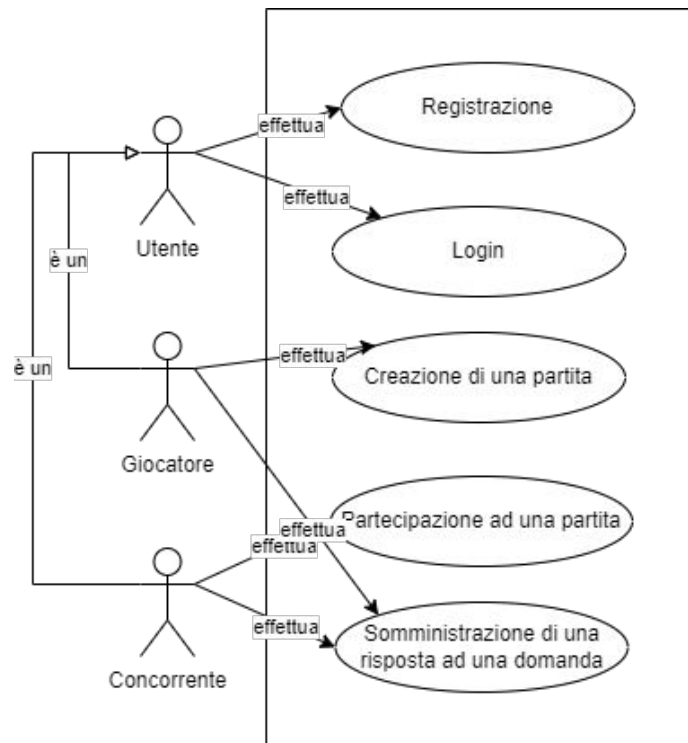
ONE VS ALL

A web based quiz game
implementation

One vs all

One vs All è un gioco multiplayer online nel quale un giocatore si ritrova ad affrontare altri concorrenti in una serie di domande a scelta multipla. I partecipanti accedono al turno successivo rispondendo correttamente alla domanda del turno corrente. Ogni turno ha la durata di 30 secondi. Una volta terminati, il turno è considerato concluso e sarà elaborato il turno successivo. Il gioco termina quando il giocatore, rispondendo correttamente a tutte le domande, vince ed elimina tutti gli avversari oppure sbaglia una risposta venendo così sconfitto.

Casi d'uso



Caratteristiche

- Fault tolerance: ripristino funzionalità in caso di errore
- Scalabilità: gestione di un altro numero di utenti contemporaneamente
- Replicabilità: riproducibile indipendentemente dall'ambiente di esecuzione

Design architetturale

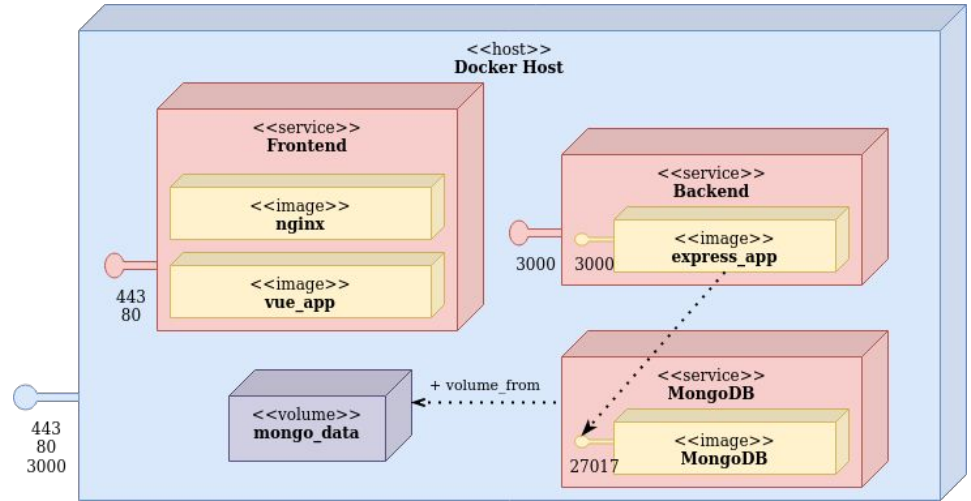
Stile architetturale ReST.

Identificazione dei servizi:

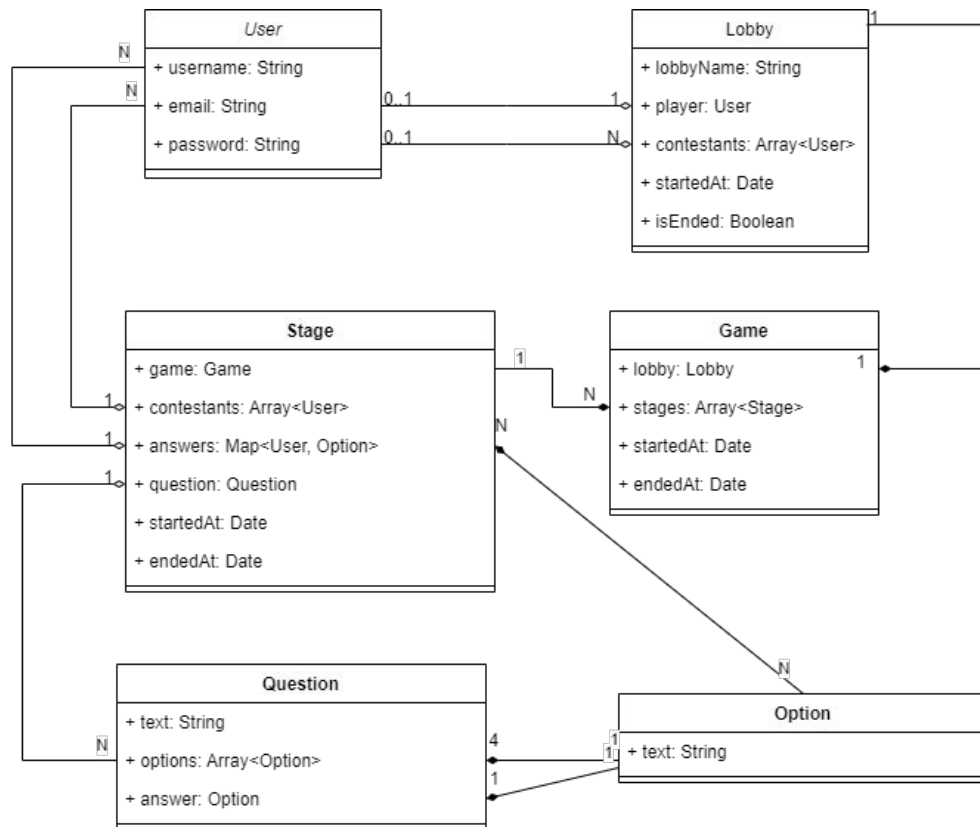
- frontend
- backend
- persistenza

Volumi ausiliari:

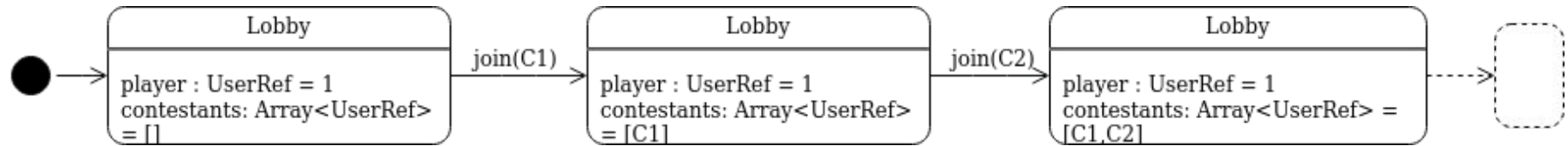
- contenuto persistenza



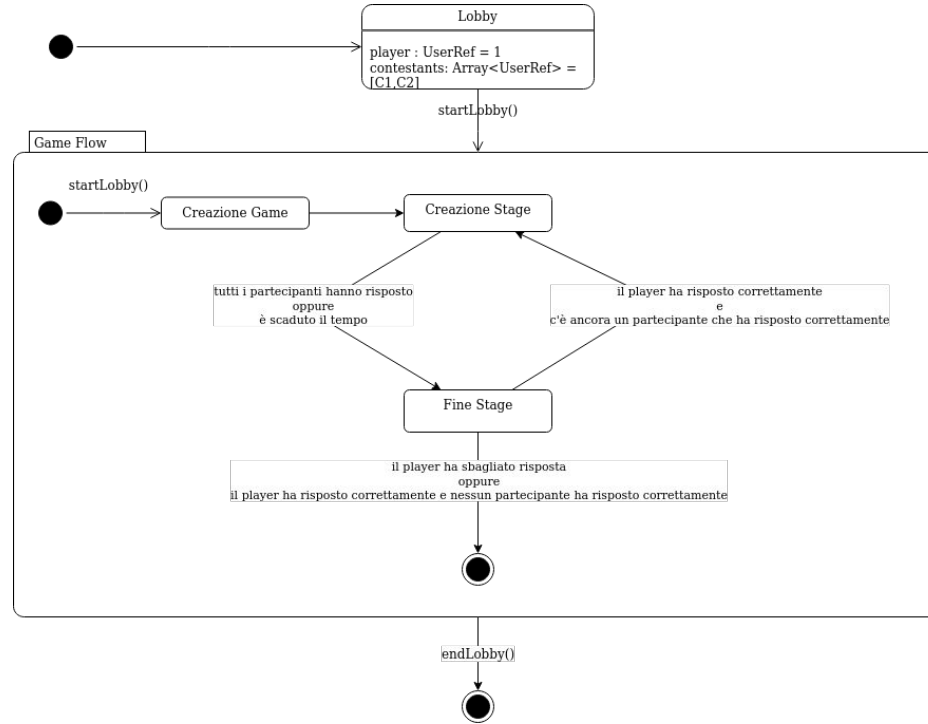
Design entità



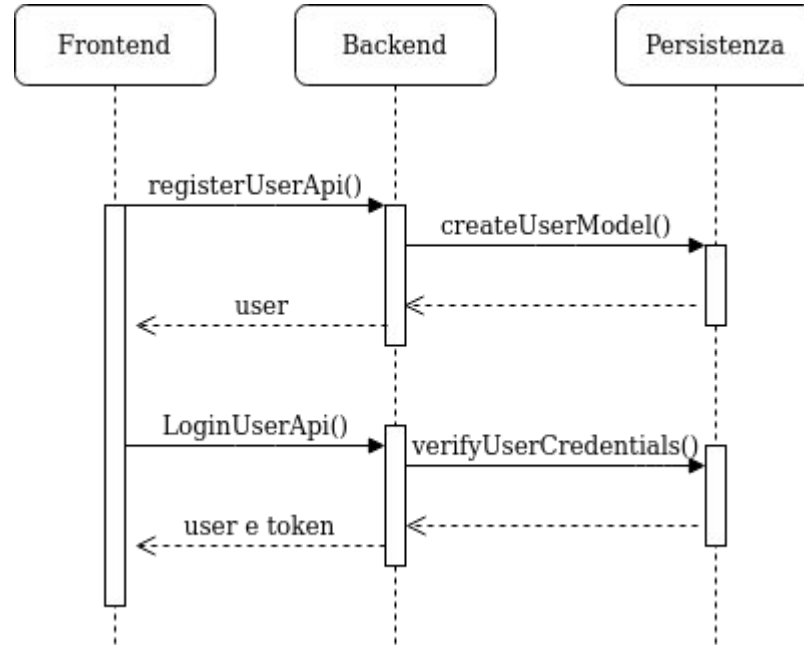
Design comportamento - unirsi ad una lobby



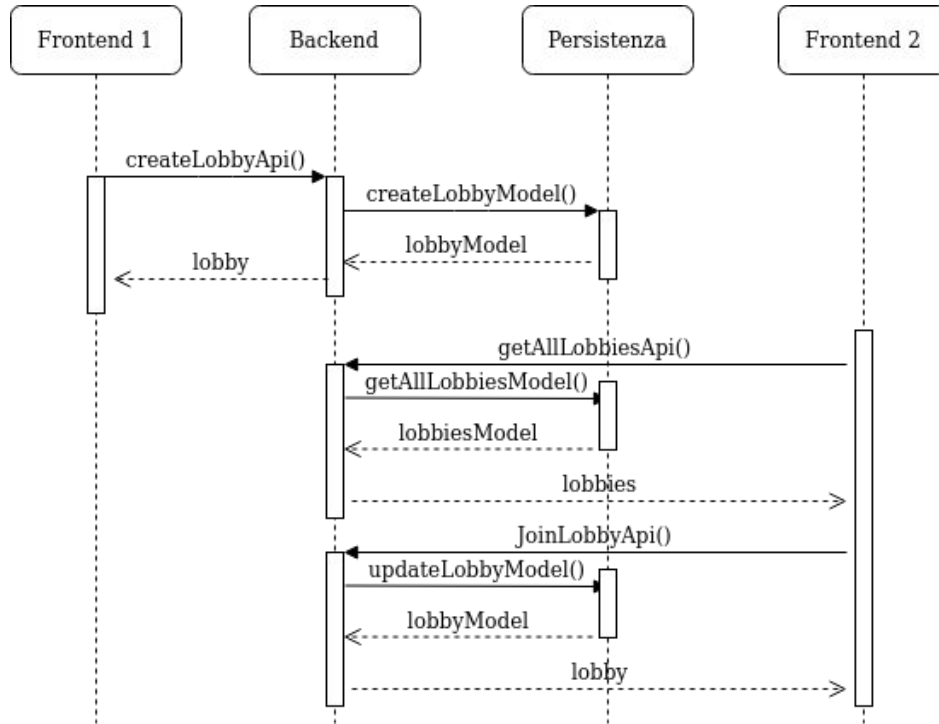
Design comportamento - flusso di gioco



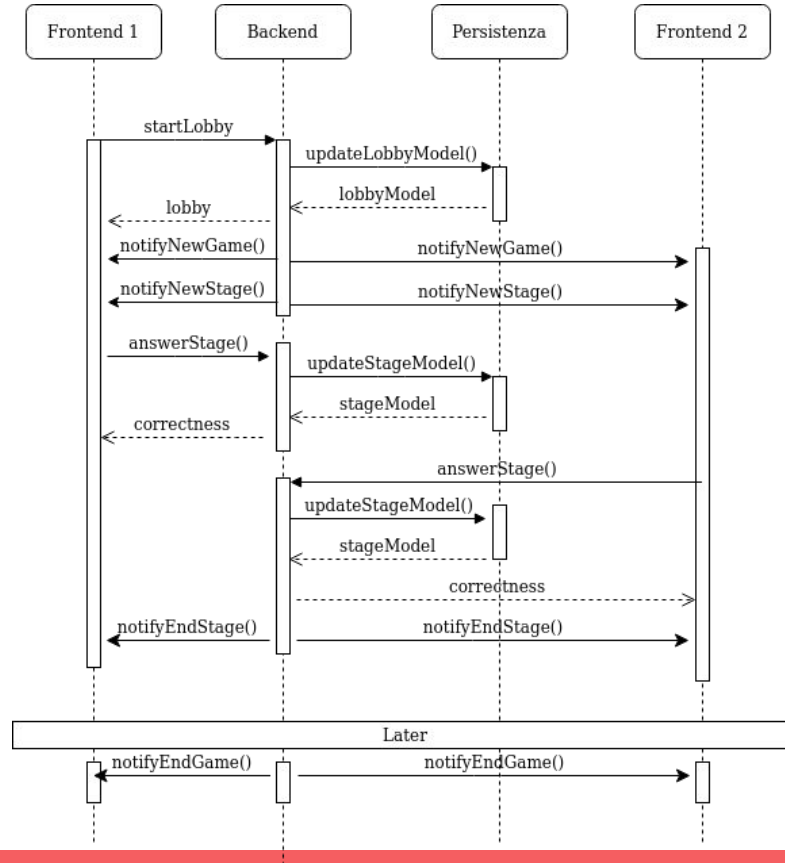
Design interazione - autenticazione



Design interazione - creazione di una lobby



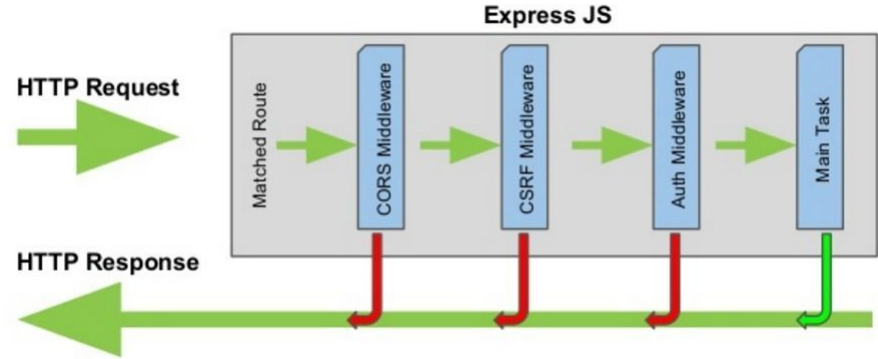
Design interazione - flusso di gioco



Implementazione - routing

Il servizio di backend utilizza una serie di middleware che validano le richieste provenienti dall'esterno.

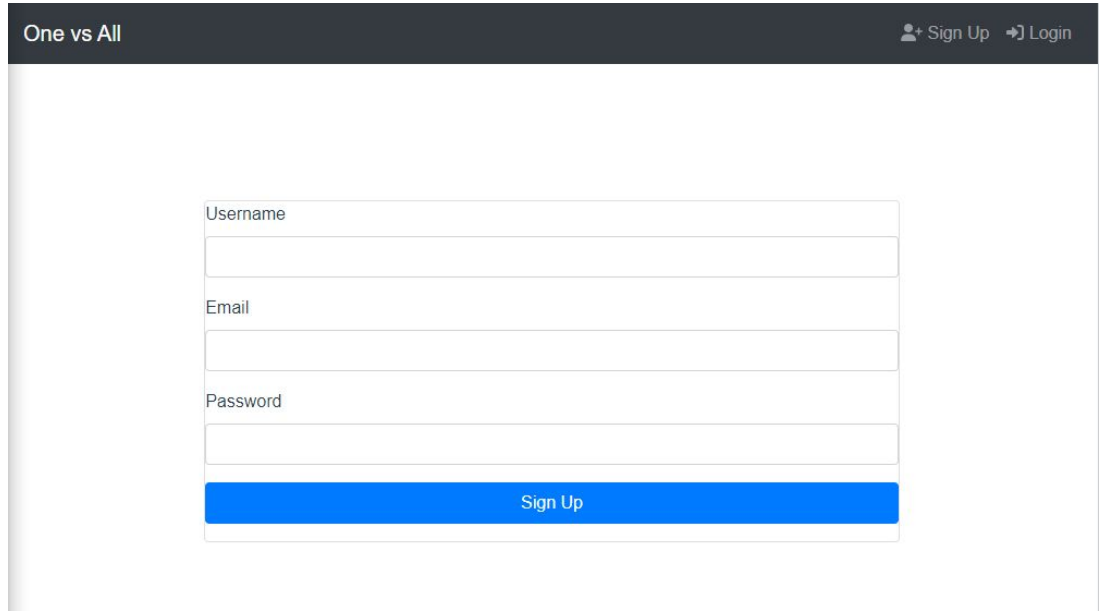
Alcuni middleware verificano che la richiesta provenga da una fonte autorizzata. Altri verificano che a livello applicativo questa sia processabile.



Implementazione - autenticazione

Al path `/api/auth/signup` è possibile richiedere la creazione di un'utenza.

Successivamente al path `/api/auth/signin` è possibile effettuare il login con le credenziali create in precedenza.



The screenshot displays a web application interface with a dark header bar. On the left of the header is the text "One vs All", and on the right are links for "Sign Up" (with a user icon) and "Login" (with a key icon). The main content area is white and contains a sign-up form. The form has three input fields labeled "Username", "Email", and "Password". Below these fields is a prominent blue button labeled "Sign Up".

Implementazione - modalità di gioco

Al path `/api/lobby/lobbyAvailable` è possibile richiedere tutte le lobby disponibili. Una volta identificata la lobby di proprio interesse è possibile partecipare tramite il path `/api/lobby/join`.

In alternativa, è possibile creare una lobby al path `/api/lobby/create`.

Successivamente, si può entrare in una lobby.

One vs All

LogOut

Lobbies Created

• lobby creata

ENTER

Lobbies Joined

• lobby creata da altri

ENTER

Lobbies Available

• lobby creata da altri 2

JOIN

Create a Lobby

CREATE

Implementazione - flusso di gioco (1)

One vs All

LogOut

Lobby name: lobby creata

Contestants: 1

Current stage: 1

Answer this question:

what singer appeared in the 1992 baseball film 'a league of their own'?

brandy

madonna

garth brooks

whitney houston

0:52

Leave Lobby

Implementazione - flusso di gioco (2)

One vs All

LogOut

Lobby name: lobby creata

Contestants: 1

Current stage: 2

Answer this question:

what mathematical term is used to describe the average of a series of numbers?

median

mode

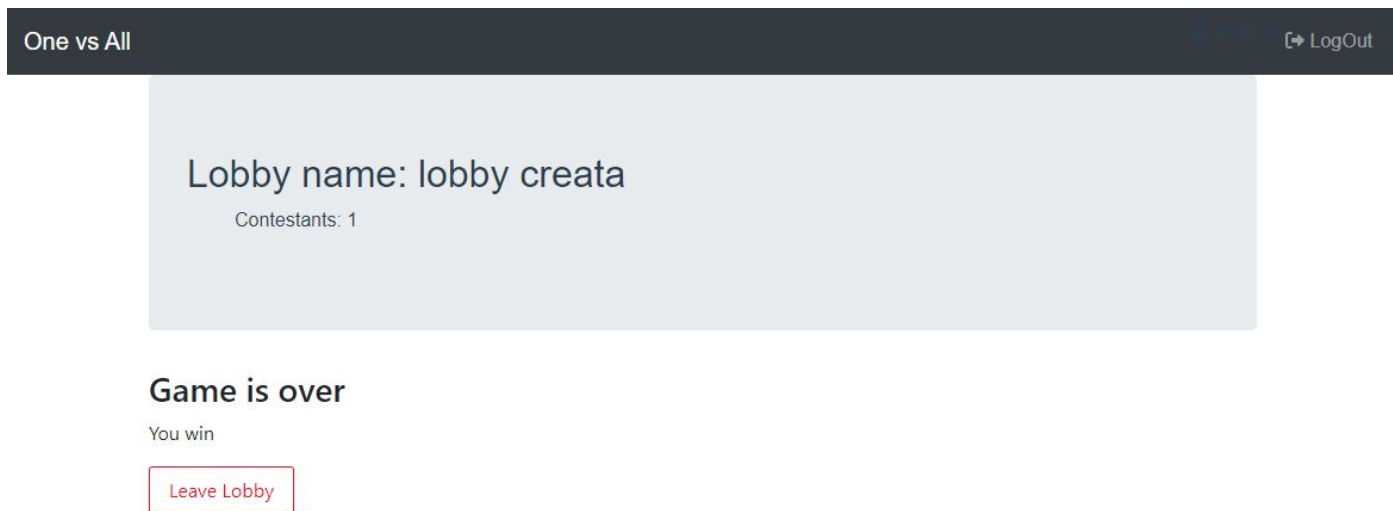
majority

mean

0:59

Leave Lobby

Implementazione - flusso di gioco (3)



Swagger API

One vs All Platform REST API Docs 1.0.0 OAS3

Authorize



Authentication



POST /api/auth/signup Register a user



POST /api/auth/signin Login a user



Lobby



GET /api/lobby/lobbyAvailable Retrieve all lobbies not started yet



POST /api/lobby/create Create a lobby



POST /api/lobby/join Join a lobby



Deploy

Tramite l'ausilio di Docker è possibile comporre lo stack di servizi che creano la soluzione presentata tramite un semplice comando.

La containerizzazione aumenta il grado di scalabilità e fault tolerance della soluzione e facilita la replicabilità di questa su un altro ambiente.

Grazie al pattern IoC, è possibile eseguire i test automatici del sistema su un'istanza di persistenza diversa da quella di produzione.

```
Test Suites: 8 passed, 8 total
Tests:      47 passed, 47 total
Snapshots:  0 total
Time:       7.846 s
Ran all test suites.
#
```

Grazie per l'attenzione