Cinema Reservation App

Krzysztof Bryszak 156052

Description:

Cinema Reservation App is a Cassandra and Python based app that simulates basic reservation process in cinema. It allows you to make, update and view reservations and also run three stress tests to check the performance.
It runs on two docker nodes that host the Cassandra clusters and on a python script in the console.
You can choose which node to connect to at the start of the script execution. Then you can:

1. Login (provide some username),
2. Exit,
3. Run Stress Test 1,
4. Run Stress Test 2,
5. Run Stress Test 3,
6. Clear All Reservations

When you choose to Login and provide an username you can:

1. Make Reservation,
2. Update Reservation,
3. View Your Reservations (for the username that you provided at login)
4. View All Reservations (for a provided movie - show_id)
5. Logout

Database Schema

```
CREATE KEYSPACE cinema WITH replication = {'class': 'SimpleStrategy', 'replication_factor': '2'}  AND durable_writes = true;

CREATE TABLE cinema.reservations (
    show_id text,
    seat_id text,
    reservation_time timestamp,
    user_id text,
    PRIMARY KEY (show_id, seat_id)
);
```

Problems encountered

1. Stress test 3: How to allow each user to make at least some reservations?

- I added a small random delay before executing a reservation query for each seat. It works most of the time and eliminates the advantage that one thread simulating a user is run earlier than the second one. Still sometimes one user gets all the reservations (bigger delays could prevent that but it will decrease the performance)

2. Application loses data when the nodes are shut down

- I used Docker Volumes to store the data locally between runs.