Shadhin Music Search Engine DOTNET Microservices: CQRS & Event Sourcing with Kafka

Description

In Search Engine, used .NET microservices that comply with the CQRS and Event Sourcing patterns.

We will not use any CQRS framework, but you will write every line of code that is required to effectively **create our own CQRS and Event Sourcing framework** using **C#** and **Apache Kafka**. We have followed CQRS and Event Sourcing in the following ways.

- Handle commands and raise events.
- Use the **mediator pattern** to implement command and query dispatchers.
- Create and change the state of an aggregate with event messages.
- Implement an **event store** / write database in **MongoDB**.
- Create a read database in **MS SQL**.
- Apply **event versioning**.
- Implement **optimistic concurrency control**.
- Produce events to Apache Kafka.
- Consume events from Apache Kafka to populate and alter records in the read database.
- **Replay the event store** and recreate the state of the aggregate.
- Separate read and write concerns.
- Structure our code using **Domain-Driven-Design best practices**.
- Replay the event store to **recreate the entire read database.**
- Replay the event store to **recreate the entire read database into a different database type - PostgreSQL.**

The ultimate goal of this project is to take a **deep-dive into the world of CQRS and Event Sourcing** to enable us to create **microservices that are super decoupled and extremely scalable**.