EVENT STORE

Training Course

An Introduction to Event Sourcing

Training Course

An Introduction to Event Sourcing

Kacper Gunia

This course is delivered by Kacper, an independent software architect, trainer, and consultant, with 6 years of experience of Event Sourcing, CQRS and Domain-Driven Design.

Target audience

Software Developers & Architects not familiar with the pattern willing to learn how to implement microservices using Event Sourcing

Structure

30% lecture and 70% hands-on coding and exercises

Duration

1 day

Number of attendees

4 - 12 with a single trainer, 13-24 with two

Course Overview

Learning a new architectural pattern is a challenging exercise as for a neophyte it can be hard to find information that is necessary to understand the foundations of the subject.

That's why learning from practitioners that know the subject well will give you a head start and help understand the steps you will have to take.

Event Sourcing is a pattern which challenges the status quo of persisting the data in microservices and distributed applications.

Instead of saving the latest state, it persists the full history of changes, which then can be used to derive the latest information.

In turn it will give you powerful capabilities such as audit trail, scalability, traceability and many others.

Description

In this training you will learn the basics of the pattern, and at the end of it you will be able to implement a simple application leveraging Command, Events, Aggregate and a Projection.

By learning about the example you will be able to understand the benefits of using Event Sourcing and learn how it relates to Command-Query Responsibility Segregation (CQRS).

You will know what are the traits of a good event store and understand in which situation the pattern is applicable and what are the topics that you should learn next.

Training Course

An Introduction to Event Sourcing

EVENT STORE

Learning Outcomes

- Explain 5 benefits of Event Sourcing
- Recognise challenges of a 3rd normal form model and explain how CQRS addresses it
- List 3 traits of an Event Store (global ordering, optimistic concurrency & fine grained streams)
- Implement an application with one Event Sourced Aggregate
- Explain the difference between Command and Event
- Implement a Projection that will create a read model

Scope

Introduction

- Challenges of classic persistence model
- CQRS and Event Sourcing
- Event Sourcing as persistence model
- What is required from Event Store

Benefits of Event Sourcing

- Audit trail
- Traceability, Debuggability & Observability
- Scalability & Fault Tolerance
- Temporal queries
- Separation of concerns & Decoupling

Building blocks

- Domain Event
- Command
- Aggregate
- Projection

Prerequisites

- Proficiency in one mainstream programming language
- Laptop/notebook with a working development environment and ability to connect to internet
- A sample web application up and running (should be able to accept an http request)
- Docker daemon up and running

Room setup

- Cabaret or boardroom style setup
- Projector / Screen
- Whiteboard / Flipchart

EVENT STORE®