

Training plan

Advanced Domain-Driven Design: Modelling, CQRS & Event-Sourcing in practice

Target audience: Software Developers & Architects

Structure: 30% lecture and 70% hands-on modelling and coding

Duration: 3 days

Description:

Modelling complex and scalable IT systems is a difficult task and there is no flowchart on how to approach such a challenge as every project has different requirements, context and teams implementing it. Domain-Driven Design is an approach that focuses on managing this complexity by aligning with the business domain to develop highly maintainable systems that deliver on business requirements.

Two technical patterns for leveraging DDD are CQRS and EventSourcing. The Command-Query Responsibility Segregation architecture is a battle-tested approach used to design extremely high scale systems by decoupling reads and writes. Event Sourcing is one possible implementation of the write stack which gives additional benefits such as a 100% reliable audit log, ability to execute temporal queries, and replays that allow to gain new insights from historical data or correct data inaccurately processed in the past.

In this training you will gain the necessary experience on how to model and implement complex systems using Domain-Driven Design, CQRS and Event Sourcing.

What you will learn:

- How to analyse business requirements and model them using Event Storming
- How to divide complex systems into smaller, manageable, domain-aligned sub-systems: Bounded Contexts
- How to decouple reads and writes for enhanced scalability with CQRS
- How to design and implement Event-Sourced microservices
- How to deal with Eventual Consistency in CQRS systems
- How to implement, monitor and replay projections
- How to meet performance and scalability requirements

Room set-up:

Cabaret style set-up with a table for each group of 4 or 5. Plenty of wall space used for modelling and hands-on exercises. Whiteboards / flipcharts. Projector.

Hardware requirements:

Each participant needs a laptop with a development environment and internet connection.

Training scope:

1. DDD building blocks

- a. Events
- b. Aggregates
- c. Commands
- d. Value Objects
- e. Entities
- f. Services

2. Domain Modelling

- a. Big Picture Event Storming
- b. Event Modelling / Design Level Event Storming
- c. Bounded Context discovery heuristics

3. CQRS & Event Sourcing

- a. Read & Write stacks
- b. Temporal queries
- c. Auditing

4. Commands & Event-Sourced Aggregates

- a. Aggregate design
- b. Snapshotting
- c. Aggregate testing using Given-When-Then formula
- d. Task-based User Interfaces

5. Queries & Projections

- a. Projection design
- b. Side-effect handling
- c. Downtime-free Replays
- d. Long-running processes
- e. Projection testing using Given-When-Then formula

6. Performance & Scalability

- a. Eventual consistency
- b. Integration via Public Events
- c. Monitoring
- d. Autoscaling
- e. Partitioning