trainitek

Applying DDD & EventStorming in Modern Architecture

This training is the perfect match for those who attended the “Implementing Modern Architecture” training and would like to know more about how Domain-Driven Design works in real life.

During 2 days, we will primarily focus on Domain-Driven Design (DDD) and EventStorming. Those tools help to align software architecture with business needs and increase the chances of an architecture being successful.

By being a part of the series of hands-on workshops, participants will learn how to understand business needs, discover bounded contexts, and design working models using EventStorming. The training covers strategic and tactical DDD, including designing aggregates and value objects, ensuring invariants, working with domain events, accessing entities using repositories, and more. It’s ideal for IT specialists who would like to build software systems that solve business problems and are well-designed from a technical perspective.

Audience: Architects, Team Leads, Developers
Duration: 2 days

- Day 1: Domain-Driven Design, Big Picture EventStorming workshop, Process Modeling with EventStorming workshop, Discovering Bounded Contexts workshop
- Day 2: Essentials of Tactical DDD, Discovering Aggregates workshop, Applying Tactical DDD in real code

Format: 50% workshop / 50% lecture

Training program

1. Domain-Driven Design
    a. What is DDD and why use it?
    b. Strategic DDD
    c. How to discover a Bounded Context?
2. Introduction to EventStorming
    a. What is EventStorming and why use it?
    b. Preparation for an EventStorming workshop
3. Big Picture EventStorming workshop
    a. Discovering events, systems, and actors
4. Process Modeling with EventStorming workshop
    a. Discovering Commands, Policies, and Read Models
5. Discovering Bounded Contexts workshop

6. Essentials of Tactical DDD
    a. Value Objects
    b. Entities
    c. Aggregates
7. Discovering Aggregates workshop
8. Applying Tactical DDD in real code
    a. Designing Aggregates and Value Objects
    b. Ensuring Invariants in a Domain Model
    c. References in Object Graphs
    d. Application, Domain, Infrastructure, and Presentation layers
    e. Implementing complex scenarios using Domain Services
    f. State transitions in Aggregates
    g. Working with Events in a Domain Model
    h. Managing Aggregates using Factory, Repository, and Memento
    i. Implementing Concurrency