DDD training

A 3-day workshop-based introduction to Domain-driven design. During this workshop attendees learn how to understand real business requirements, discover a domain, design a business model and finally implement a working solution based on an example business domain.

Example project will be implemented in Java and Spring Framework.

Requirements: 4+ years of experience in Java development (or other OOP language), good understanding of OOP design

Training Outline

1. Introduction to DDD
    a. What is domain
    b. DDD approach
    c. When use DDD
    d. When not use DDD
2. Model vs domain
3. Ubiquitous Language
4. Anemic model
5. Domain modeling
    a. Knowledge Crunching
    b. Event Storming
    c. Example Mapping
    d. BDD/TDD in DDD
    e. Deep Model
    f. Breakthrough
6. Communication
    a. Ubiquitous Language as a tool for business and developers
    b. Modeling session example
7. Tactical patterns - code examples
    a. Entities
    b. Value objects
    c. Services
    d. Modules
    e. Aggregates
    f. Factory
    g. Repository
8. GoF patterns in DDD
9. Supple design
10. Strategic patterns

    a. Types of domain
        i. Core
        ii. Supporting
        iii. Generic
    b. Bounded Context
    c. Bounded Context vs module vs microservice
    d. Bounded Contexts integration
    e. Context Map
    f. Communication between teams working on different bounded contexts
    g. Wardley mapping
    h. Core domain charts
    i. Business model canvas

11. DDD in architecture
    a. DDD impact on architecture
    b. Architectural patterns
        i. Layered Architecture
        ii. CQRS
        iii. Event driven architecture
        iv. SOA
        v. Hexagonal Architecture
12. Large scale structure
    a. Evolving order
    b. Responsibility layers
    c. System metaphor
    d. Knowledge level
13. Domain discovery techniques
14. Introduction to example domains
15. Implementing project based on example domains - workshop