Sprint 1.MSD: Business Domain Design

Sprint_1 •

Overview

Business Components Design:

- **Business Requirements and/or Business Workflow**
- **Domain Model and Business Rules**

1. Use Cases
Business Workflow Rules

Business Requirements

- Use Cases

- (or Usability Scenarios)

- (or Business Stories)

2. Domain Model

Business Entities

- Entities and associations

- ORM Model

Entity Aggregates

- Root Entity

- Components

  - Component Entities

  - Value Objects

Business Rules

Validation Rules

- Entity Fields Validation rules

- Aggregate Validation rules

Computation Rules

- Computed Entity Fields

- Computed Entity Views

2. Deliverables

Business Requirements

Use Cases Diagrams

* (and/or) Sequence Diagrams

* (and/or) Activity Diagrams

Domain Model

Entity Class Diagrams

(extra) SQL Diagrams

(extra) Business Rules Specifications
(non-formalized/textual form)

Sprint 2.MSD: Business Model Implementation

Sprint_2 •

Overview

Business Components Implementation:

- **Domain Model Implementation**
- **Business Rules Implementation**

Course Coverage:

- **Theme_2: Course_3, Course_4**

Domain Model Implementation

Domain Entities

- Entities and relationships

- Value Objects:

  - Enums

  - VO Custom Data Types

Entity Aggregates

- Root Entity

- Components

  - Component Entities

  - Value Objects

- Associations within aggregates

Business Rules Implementation

Business Workflow Rules

- Business Workflow (Use-case) Services

Validation Rules

- JavaBean Validation Rules annotations into Domain Entities

- Domain Validation Services

Computation Rules

- Domain Computation Services for Entity *Fields*

- Domain Computation Services for Entity *Views*

Deliverables

Entity **Classes**

Entity Support Services **Interfaces** and Implementation **Classes**

- Aggregate Entity Factories

Business Services Classes **Interfaces** and Implementation **Classes**

- **Workflow Services**
- Validation Service
- Computation Services
- Audit Services

Spring App Configuration **Class**

JUnit Test **Classes**

- Test for Validation Service
- Test for Computation Services
- Test for Workflow Services

Sprint 3.MSD: Persistence Model Implementation

Sprint_3 • 2021

Overview

Data Access Components Implementation:

- **Persistence Model Implementation**
- **Repositories Services Implementations**

Course Coverage:

- **Theme_2: Course_5**

Persistence Model Implementation

JPA Entities

- Entities with
  - @Entity
  - @Inheritance

- Value Objects:
  - @Enumerated
  - @Convert

JPA Aggregates

- Root Entity relationships. e.g.: @OneToMany
  - mappedBy
  - fetch
  - Cascade = CascadeType.ALL
  - orphanRemoval = true

- Component Entities relationships: @OneToMany

Repository Services' Implementation

Spring Repositories

- Extending Spring Interface

  - JpaRepository<RootEntity, T>

- Adding custom operations, e.g.

  - findByPropertyName()

  - findByCustomFilter() with @Query to add custom filter

(Bonus) Complete JPA Repository with

- IRepository CRUD interface

- EntityRepositoryJPA implementation class with EntityManager injection as @PersistenceContext

Business Rules Implementation

Update existing Business Services and Factories

- to use JPA-based Repositories, either Spring Repositories or complete JPA repositories

Implement computation or audit rules

- using @EntityListeners classes with @PrePersist, @PreUpdate, @PreRemove methods.

Deliverables

persistence.xml

@Entity **Classes**

Repository definitions as

- **Spring JpaRepository interfaces**

(Bonus) Repository definitions as

- EntityRepository CRUD interfaces with
- EntityRepositoryJPA implementations

Spring App Configuration **Class** with JPA beans' factory methods for:

- EntityManagerFactory and
- JpaTransactionManager

JUnit Test **Classes**

- New tests for JPA Repository Services
- Updates Tests for Business Services