

Formation Domain Driven Design (DDD) | Appréhender les principes d'une conception orientée Métier

Appréhender les principes et les pratiques d'une conception orientée Métier

Fiche en date du 29/01/2023. ([Voir sur le site](#))

Informations pratiques



[Voir nos formateurs](#)



3 rue de Tolbiac, 75013
[Voir le plan](#)



3 jours



1 850 €

Prochaines sessions :

Contactez-nous à training@soat.fr

Nombre de participants maximum : 10

Objectifs

- Comprendre les bénéfices du Domain Driven Design (DDD)
- Savoir distinguer la complexité technique de la complexité de son métier
- Être en mesure de prendre la décision d'appliquer ou non DDD à son contexte
- Permettre d'aligner les concepts DDD avec ses propres pratiques issues de son expérience
- Maîtriser et appliquer les patterns DDD stratégiques
- Connaître les techniques d'introduction du domaine dans le code
- Avoir une expérience d'implémentation (architecture + code) concrète de DDD
- Connaître et utiliser les principales briques DDD tactiques

Public

- Architecture, Développeur confirmé (ayant une bonne expérience, idéalement de plusieurs années sur des domaines complexes)

Prérequis

Connaissances d'un langage orienté objet

Méthodes pédagogiques

40 % théorie / 60 % pratique

Description

Depuis ces dernières années, la complexité des problèmes métiers ne cessent de s'accroître, et très souvent, les applicatifs ne répondent pas correctement aux besoins des utilisateurs. On s'aperçoit ainsi que le but ultime n'est pas d'écrire du code ou même de construire des logiciels mais principalement de résoudre des problèmes. Pour répondre à la construction de domaines métiers complexes, DDD est une philosophie fournissant un ensemble de pratiques, de principes et de patterns pour dé-complexifier le métier et produire des applications répondants mieux aux besoins des utilisateurs, et ceux quel que soit le langage de programmation ou l'environnement technologique.

Cette formation Domain Driven Design (DDD) vous permettra de mieux comprendre le vaste et complexe sujet DDD. Tout au long de la formation, un fil rouge applicatif est co-construit afin de comprendre la mise en œuvre des différents principes, pratiques et patterns d'une conception dirigée par le métier.

Programme

Introduction

- Les challenges de la création logicielle
- Une première définition de DDD
- Plus de 10 ans déjà !
- Pourquoi s'intéresser au DDD ?
- À quoi ressemble une application conçue avec DDD ?
- Les conditions pour appliquer DDD
- Les prérequis à sa mise en place

Analyse et exploration du métier

- Techniques de collaboration avec les experts métiers
- Création d'un « ubiquitous language »
- Focus sur le « Event Storming »
- Expérimentation du « design » émergeant

Modèle et Bounded Context

- Définition et application d'un mode
- Différents types de modèles possibles : Anemic, Transaction Script, CRUD
- Le « Rich Domain model »
- Subdivision et catégorisation en sous-domaines
- Notion de « Bounded context »
- Création d'un « Navigation Map »
- Maintien du « Domain Model » avec le « Bounded Context »
- Mise à jour du Domain Model
- Context Mapping
- Intégration des « Bounded Context »

Patterns de modélisation

- Application de l'hexagonal Architecture
- Utilisation des building blocks
- Entity, Value Objects, Domain Services et Modules
- Aggregats, Repositories et Factories
- Focus la séparation du « Data Model » vs « Domain Model »
- Règles de validation & Spécifications

Évaluation

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de quizz, mises en situation, travaux pratiques... En fin de formation, il est également demandé aux participants de mesurer leur satisfaction vis-à-vis de de la formation suivie. SOAT Training dispose d'un processus qualité qui prend en considération les éventuels dysfonctionnements rencontrés par les participants afin d'être proactif quant à la solution corrective adaptée tant sur le contenu de la formation elle-même que les conditions de son déroulement.