

## Le Domain-Driven Design par la pratique (2)

Le Domain-Driven Design ou "DDD" est un incontournable dans la carrière d'un développeur.

Cette formation s'adresse aux développeurs confirmés.

En construisant une application exemple, les principes, les patterns tactiques et stratégiques du DDD sont abordés de manière concrète, ainsi que des sujets comme l'Event Sourcing ou même le DDD et la programmation fonctionnelle.

Formation en Java.

Cette formation s'inscrit dans le cadre d'un parcours DDD :

- AE-IDDD – Introduction au Domain-Driven Design [1 jour]
- AE-DDD1 – Le Domain-Driven par la pratique (1) [2 jours]
- AE-DDD2 – Le Domain-Driven par la pratique (2) [2 jours]

### Détails

- **Code** : AE-DDD2
- **Durée** : 2 jours ( 14 heures )

#### Public

- Développeurs expérimentés
- Leaders Techniques

#### Pré-requis

- Avoir suivi la formation Le Domain-Driven Design par la pratique (1)

Avoir une expérience en programmation Objet si possible Java ou C# (les exemples sont en Java, il est possible de faire la formation en C#)

### Objectifs

- Explorer de nouvelles pratiques de code et d'architectures liés au DDD
- Découvrir l'Event Sourcing, le DDD et la programmation fonctionnelle

### Programme

#### L'IC version DDD

- L'Intégration continue des concepts, le refactoring de concept
- Dérouler un Scénario sur le modèle de notre application fil rouge

#### Recap : les moteurs du DDD

- UL et BC
- Un moteur d'évolutions (agile) : UL/Code/Modèle/IC
- NY accélère NY

#### Les agrégats, le cœur du problème

- Pourquoi c'est si important ?
- Et avec plusieurs agrégats ?
- Les VO ID
- Les événements de domaine
- Eventual consistency
- Les Services de domaine
- L'injection pour découpler

#### Mise en place sur notre application

#### Le Context Mapping

- Les patterns
- Un exemple sur notre application fil rouge
- Détail sur le pattern BBOM

#### CQRS et Event Sourcing

- La dualité état / événement
- Les principes d'architecture CQRS
- Attention au Command sourcing
- Le cœur de l'Event Sourcing
- Framework d'ES

#### Le DDD, l'approche objet, l'approche fonctionnelle

- La pensée objet et la pensée fonctionnelle
- Le DDD et le fonctionnel
- Exemple en Clojure

#### Conclusion

- Le DDD, vers une maturité du SI
- Le DDD, ne sera jamais dogmatique
- L'avenir du DDD

**Répartition : 40% théorie / 60% de pratique**

## Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle – 1 poste par stagiaire – 1 vidéo projecteur – Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés – Cas pratiques – Synthèse
- **Validation** :Exercices de validation – Attestation de stages