

Mis à jour le 25/10/2023

S'inscrire

# Formation Architecture Hexagonale et TDD

3 jours (21 heures)

## Présentation

L'Hexagonal Architecture n'est pas une architecture à proprement parler, mais des principes d'architectures mis en avant par Alistair en 2005 mais aussi affinés par tout un tas d'autres auteurs dont un principal : Robert C. Martin alias Uncle Bob en 2012.

Ce dernier a intitulé sa variante de l'hexagonale nommée « Clean Architecture » en apportant pas mal de guidelines et de prévention.

Ces principes d'architecture permettent tout un tas de bénéfices :

- Découplage fort entre la partie métier de l'application et la partie infrastructure/technologique, rendant la conception du cerveau de l'application bien plus facile
- Testabilité de la partie métier grandement améliorée ; le réel TDD rendu possible !
- Capacité à remettre à plus tard les choix technologiques d'infrastructure.
- Capacité à changer les technologies sans effort, évitant alors des refontes chronophages et fastidieuses

À l'issue de cette formation Architecture Hexagonale, vous apprendrez à maîtriser les principes essentiels, produire des logiciels organisés et créer des modèles de domaines performants.

## Objectifs

- Maîtriser les principes de l'Hexa/Clean Architecture, tels que l'inversion de dépendances
- Savoir démarrer un projet from scratch pratiquant TDD et l'Hexa Architecture avec l'esprit d'"émergence de design du code"
- Sensibilisation aux décisions cruciales d'architecture technique globale
- Savoir manier un TDD orienté comportement dans une Hexagonal Architecture pour une productivité accrue et sans faille
- Savoir intégrer des composants d'infrastructure tels qu'une base de donnée PostgreSQL et des API tierces partenaires sans toucher au coeur de l'appli
- Savoir dissocier la logique métier de l'application du framework Spring-Boot

- Connaître tous les pièges à éviter dans une Hexa/Clean Architecture
- Être pleinement conscient de la différence majeure entre TDD et le simple fait d'écrire des tests
- Savoir ordonner les actions à réaliser lors de l'élaboration d'une Hexagonal Architecture
- Toutes les questions répondues et quiproquos/déformations populaires au sujet des pratiques révélés

## Public visé

- Technical Leaders
- Développeurs Backend
- Développeurs Full Stack
- Architectes techniques

## Pré-requis

- Maîtrise de Java ou tout autre langage orienté objet
- Notions des concepts principaux de la POO : Interfaces / Classes abstraites / Polymorphisme
- Notion d'écriture de tests avec JUnit 5 et AssertJ

## Technologies utilisées

- Java 20
- Maven 3
- Spring-Boot / Rest APIs
- Hibernate/JPA
- PostgreSQL
- JUnit 5 / AssertJ
- TestContainers (Docker)

## Programme de notre Formation Architecture hexagonale et TDD

### Jour 1 : Fondations et Pratiques

#### Analyse des architectures en couches classiques

- Introduction aux problèmes courants
- Démonstration par l'exemple des pièges et difficultés

*Comprendre les limites actuelles pour aspirer à une meilleure approche*

## Plongée dans l'inversion de dépendances

- Magnificence du "D" de SOLID
- Application pratique pour optimiser un code

*En utilisant l'inversion de dépendances, on crée un pont naturel vers l'Hexagonal Architecture*

## Transition vers l'Hexagonal Architecture

- Historique : origines et évolution
- Comparaison avec Clean Architecture et Onion Architecture
- Exploration des mécanismes clés

*Une fois cette base solide établie, comment garantir une implémentation efficace ? Le TDD*

## Immersion dans le Test-Driven Development (TDD)

- Briser les idées reçues sur TDD et tests unitaires
- Challenges interactifs pour tester la compréhension
- Introduction aux différents types de doublures

*Avec le TDD, nous pouvons pratiquer le codage de manière éclairée*

## Atelier de codage en direct

- Initiation au Behavior-Driven Development (BDD)
- Session de codage en Java 20
- Accent sur le refactoring guidé par les tests du TDD
- Astuces pour le refactoring des tests

## Jour 2 : Infrastructure et Tests Avancés

*Entrons dans l'aspect technologique du projet*

### Mise en place de l'infrastructure

- Création de contrôleurs avec Spring-Boot
- Configuration du Container IOC de Spring
- Intégration PostgreSQL avec Hibernate
- Utilisation de TestContainers pour la base de données

*Avoir une bonne infrastructure est essentiel, mais savoir la tester est crucial*

## Panorama des différents types de tests

- Distinctions entre chaque type de test

## Atelier Tests d'Intégration et End-to-End

- Écriture "test-first" avec outils/frameworks
- Développement de tests end-to-end et d'intégration

*Intégrons des outils et services tiers tout en gardant un découplage cœur/outils*

## Exploitation de technologies complémentaires

- Intégration avec Google Maps API

## Suite de l'atelier pratique

- Continuation du live coding

## Jour 3 : Approfondissement et Extension

*Consolidons nos bases et explorons des techniques collaboratives*

## Affinement des règles de gestion

- Atelier pour affiner les règles métier

*Pénétrons dans les nuances avancées de l'Hexagonal Architecture*

## Exploration de la "Nested" Hexagonal Architecture

- Architecture Hexagonale dans un adaptateur

## Atelier d'extension de l'application

- Challenge pour les participants
- Quiz d'évaluation en live

## Session Questions/Réponses

- Discussion ouverte sur tous les concepts

## Pour aller plus loin

## Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

## Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

## Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

## Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

## Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.