



## **AGILE DEV PRACTICES**

**FORMATION - V.1 du 24/01/2024**

### **OBJECTIFS**

- ✓ Développer des réflexes de découverte et d'analyse de code (checklists, analyse static de code, linters, qualités des tests, hotspots analysis, ...)
- ✓ S'améliorer dans les pratiques collaboratives intra-équipe (Pair programming, Mob Programming ) mais aussi inter-équipes ( Example Mapping ).
- ✓ Maitriser les concepts liés au Clean Code et Clean Testing,
- ✓ Pratiquer le Test-Driven Development et comprendre qu'il ne s'agit pas d'une approche de test mais d'une approche de conception.
- ✓ Être capable de refactorer et améliorer n'importe quel code complexe.
- ✓ Comprendre comment combattre au quotidien la dette technique,
- ✓ Maitriser la Clean Architecture (Use Cases, Port / Adapters).
- ✓ Maitriser les patterns tactiques liés au Domain Driven Design.
- ✓ Comprendre comment écrire du code plus robuste à l'aide des concepts de la programmation fonctionnelle.
- ✓ Apprendre à s'appuyer sur les Gen AI dans notre quotidien (quand et pour quoi ?)

## PUBLIC CONCERNÉ

Développeurs, Technical Leaders,  
Architectes

## PRÉ-REQUIS

- Connaître les bases de la Programmation Orientée Objet.
- Maîtriser un langage orienté objet comme Java ou C#.
- Être capable d'écrire et d'exécuter des tests unitaires.

## DURÉE DE LA FORMATION ET MODALITÉS D'ORGANISATION

**Durée totale** : 2 jours

*En présentiel ou en distanciel*

**Nombre de participants** : 4 à 12

**Horaires de la formation** :

Généralement 9h – 17h00 avec une pause déjeuner.

Les horaires seront précisés aux participants en amont du cours.

## MODALITÉS D'ASSISTANCE

**Pédagogique et technique** : Yoan Thirion, formateur et Coach Agile chez Pyxis Suisse.

### Contact administratif

Carole GODEL, coordinatrice administrative de formation

Plus d'information sur : [www.pyxis-suisse.ch](http://www.pyxis-suisse.ch)

Contactez-nous : [info@pyxis-suisse.ch](mailto:info@pyxis-suisse.ch)

## CONTENU DE LA FORMATION

### Jour 1

- Tour de table et recueil des attentes
- Kata d'introduction à Test-Driven Development et Example Mapping
- Se faire une idée de la base de code : découverte du système du Bouchonnois et de ses contraintes
- Commencer par traiter les warnings comme des erreurs (Politique 0 Warning)
- Jauger la qualité des tests
- Améliorer la lisibilité / maintenabilité des tests automatisés

### Jour 2

- Simplifier une suite de tests et renforcer sa confiance vis-à-vis de notre filet de sécurité
- Mettre en place des tests d'Architecture (automatisés)
- Introduction théorique et mise en application des concepts derrière la
- Clean Architecture
- Application du principe Tell Don't Ask et renforcement de l'encapsulation de notre Domain model
- Combattre la Primitive Obsession
- Éviter les exceptions et rendre le code référentiellement transparent (Referential transparency)
- En option : transformer notre système afin de le rendre "event-sourcé" (Event Sourcing)

## MOYENS TECHNIQUES ET PEDAGOGIQUES

- **Installations et matériel requis** :
  - ✓ Tableau(s) blanc(s) / Tableau(s) papier(s).
  - ✓ Vidéoprojecteur.
  - ✓ Ordinateurs des participants (au moins 1 pour 2) avec votre IDE préféré.
  - ✓ Accès à Internet : accès libre à github / miro
- **Code katas en C# on .NET 6 et java (>=11)**
  - ✓ Les développeurs.es devraient au moins se sentir à l'aise avec un de ces langages.
  - ✓ Les participants.es pourront obtenir de l'aide pendant les exercices, mais nous n'aurons pas le temps de couvrir les bases de l'installation, de la compilation, de la syntaxe, etc.

### DESCRIPTION DE LA FORMATION

Une formation mêlant théorie et pratique dans laquelle les participants découvriront et pratiqueront de nombreuses techniques et outils permettant de délivrer du logiciel de qualité de manière itérative et incrémentale.

Cette formation vise à aider les développeur-ses à devenir meilleurs dans la conception, l'écriture et la maintenance de code dans un environnement agile.

### Modus Operandi

Les participants expérimenteront "l'apprentissage par la pratique" à travers différentes activités, le tout dans la bonne humeur et dans le respect de chacun-e :

- Ateliers permettant de stimuler l'intelligence collective.
- Itérer sur un système existant en Mob Programming que nous ferons évoluer à chaque apprentissage.

Bien que le code du système soit disponible en java et C#, l'ensemble des concepts abordés sont agnostiques du langage utilisé au quotidien.

Vous pourrez ainsi les utiliser quel que soit votre stack de développement.

### Concepts Couverts

- Agilité et "Software Craftsmanship" : l'état d'esprit associé (humilité, apprentissage continu, règle du boy scout)
- Clean Code
- Clean Testing (Styles de tests, Test Data Builders, Approval Testing, Property-Based Testing)
- Analyse de code : static / comportementale
- Example mapping
- Test-Driven Development
- Techniques de refactoring: Strangler Pattern, Automatisation, Mikado
- Mutation Testing
- Pair / Mob Programming
- Test d'architecture
- Programmation fonctionnelle (Monads et autres functors)
- Clean Architecture
- Architecture Decision Records
- Event-Sourcing (en option)

### Les bénéfices

Ce que vous êtes en droit d'attendre après cette formation :

- Amélioration de la qualité du code :
  - Réduire le nombre de bugs
  - Meilleure fiabilité des applications
  - Diminution des coûts de maintenance
- Augmentation de la productivité de l'équipe avec :
  - Test-Driven Development (TDD)
  - Refactoring Automatisé

## Pyxis Suisse

rue du simplon, 37

B.P. 326, 1001 Lausanne

Courriel : [info@pyxis-suisse.ch](mailto:info@pyxis-suisse.ch)



- Domain-Driven Design (DDD)
- Réduction des coûts à long terme
- Adaptation aux tendances actuelles et futures
- Amélioration de la collaboration et de la communication
- Réduction des risques
- Meilleure adoption des principes d'amélioration continue

### Après la formation

Toutes les ressources utilisées lors de la formation vous seront fournies et vous serez en mesure de les utiliser dans vos contextes.

En plus de ces 2 journées de formation, nous vous offrons une session d'1 heure en ligne en individuelle afin de vous coacher sur la mise en place des pratiques et techniques couvertes au sein votre contexte.

### MODALITES ET DELAIS D'ACCES

L'inscription est réputée acquise lorsque le paiement est effectué, soit par virement soit par paiement en carte bancaire suite à la réception de la facture. Les inscriptions à la formation sont clôturées sept jours ouvrés avant le premier jour du cours.

### ACCESSIBILITÉ

Notre équipe s'engage à accueillir sans discrimination les personnes en situation de handicap et à leur garantir l'égalité des droits et des chances pour accéder à l'emploi et à la formation.

Afin que nous prenions les dispositions nécessaires pour vous garantir un accueil adapté et une expérience optimale, merci de contacter Carole GODEL sur [info@pyxis-suisse.ch](mailto:info@pyxis-suisse.ch) ou par téléphone au +41 22 575 20 75.

### GARANTIE ET SATISFACTION

Si vous ne repartez pas en ayant appris quelque chose de nouveau ou si vous n'êtes pas complètement satisfait du service, vous pouvez obtenir un remboursement complet.

Aucune question ne vous sera posée.

Questionnaire d'évaluation de la satisfaction des apprenants en fin de formation.