



# Sensibilisation au Software Craftsmanship

Durée 2 jour(s) (SENSIBILISATION-CRAFT-02)

Concepts de base du Software Craftsmanship avec ateliers

## Description

Connaître les principes et les pratiques du mouvement Software Craftsmanship.

### Formation animée en présentiel

La formation en présentiel se déroule sur des jours consécutifs

### Formation disponible en mode "formation à distance"

La formation à distance se déroule de préférence sur des jours consécutifs (contactez nous si besoin de décomposer en demies journées) En inter-entreprises, l'outil de visio-conférence privilégié est Microsoft Teams

En intra-entreprises, on privilégie Zoom mais Microsoft Teams est également proposé

## Objectifs

- Avoir une idée claire de ce qu'est le Software Craftsmanship et de ses bénéfices.

### *Public*

- Développeurs
- Chefs de projets

### *Prérequis*

Aucun

### *Répartition*

50% Théorie, 50% Pratique

## Evaluations des acquis

L'évaluation des acquis de la formation se fera en séance au travers d'ateliers, d'exercices et/ou de travaux pratiques. Dans le cas d'une formation officielle éditeur, veuillez nous consulter afin que nous vous fassions part des modalités d'évaluation.

A l'issue de la formation, vous sera transmis une évaluation à chaud de l'action de formation qui vous permettra de nous faire part de vos retours quant à votre expérience apprenant avec Zenika.

## Ressources pédagogiques

Les ressources pédagogiques proviennent de productions des équipes Zenika et/ou de la documentation éditeur dans le cas d'une formation "Officielle". Les documents sont en français ou en anglais.

## RQTH et ma formation Zenika

Si vous êtes sujet à un handicap, prenez contact avec nos équipes pour que nous puissions définir ensemble comment nous pourrions aménager la session afin que vous puissiez vivre une expérience en formation inchangée.

## Programme

# Définition et chronologie du Software Craftsmanship

- Définition
- Chronologie
- Agilité et Software Craftsmanship
- Manifeste
- Attitude du Craftsman

## ROI du Software Craftsmanship

- La qualité
- La dette technique
- ROI des tests
- ROI du refactoring

## eXtreme Programming

- Histoire
- Les pratiques
- Les valeurs

## Refactoring, Clean Code et patterns

- Code smells
- Refactoring
- Patterns
- Principes SOLID

## Test Driven Development

- Définition du TDD
- Pourquoi le TDD
- Le cycle de TDD

## Autres pratiques

- Coding Dojo
- Domain Driven Design
- BBL, Book Clubs etc...
- Craftswap
- Pet projects

## Ateliers

- Bowling Kata - TDD en ping pong programming
- Refactoring - Refactoring de code legacy
- Code Review - Individuer les bad smells dans le code