

# Conception avec les Design Patterns

5 j (35 heures)

Ref : UMLP

## Public

Architectes, développeurs, analystes, chefs de projets...

## Pré-requis

Niveau : Posséder une solide connaissance en programmation orientée objet (Java)  
Techniques (formations en classe virtuelle) : Vous devez disposer d'un ordinateur connecté à internet, d'un micro et d'une caméra

## Moyens pédagogiques

Modalité : Formation présentielle ou Formation distancielle (classe virtuelle) - Inter / Intra - Groupes de 4 à 12 stagiaires

Méthodes : Présentation des concepts, discussion technique, démonstrations, exercices et TP

Matériel :

*Présentiel* : Un poste informatique par stagiaire connecté à internet, à une imprimante en réseau et au réseau informatique,

Les salles sont équipées d'un tableau interactif ou d'un vidéoprojecteur et d'un paperboard

*Distanciel* : Aelion met à disposition de chaque stagiaire

- Un PC équipé des outils et logiciels nécessaires à la formation auquel vous accédez via un outil de prise en main à distance

- Un accès à un outil de classe virtuelle (Meet)

Support de formation : Un support de formation sera remis à chaque stagiaire en fin de formation : plateforme collaborative intégrant le code source des exercices réalisés en formation, webographie, mémos

## Modalités de suivi et d'évaluation

Questionnaire d'évaluation des pré-requis, suivi des connaissances tout au long de la formation, Evaluation des acquis en fin de formation

Questionnaire d'évaluation de la satisfaction en fin de stage, feuille de présence émarginée par demi-journée par les stagiaires et le formateur, Attestation de fin de formation

Les patrons de conception (Design Patterns) font partie aujourd'hui des pratiques indissociables de tout développement orienté objet. Qu'il s'agisse de Java, PHP, C++, tous ces langages s'appuient sur des modèles de conception dont l'objectif est de rendre plus fiable les applications développées. Maîtriser les concepts et les principaux Patterns (de comportement, de création, de structuration) permet de consacrer votre ressource à la résolution des processus métiers et augmenter la performance de vos équipes. Cette formation se concentre sur les différents Patterns et patrons de conception, et précise leurs utilités respectives et leur modélisation UML. A l'issue de la formation, vous serez capable d'utiliser les principaux modèles de conception disponibles (Design Patterns).

Action collective OPCO ATLAS - [Inscription CampusAtlas](#)

## Objectifs

Conception avec les Design Patterns

Expliquer la philosophie des Design Patterns

Définir et implémenter les principaux patterns

Appréhender la démarche d'utilisation des modèles pour appliquer le bon patron

Appliquer les Design Patterns dans une architecture technique de conception

## Programme détaillé

### **EXPLIQUER LA PHILOSOPHIE DES DESIGN PATTERNS**

---

Principes fondamentaux de conception

Les Patterns Grasp d'affectation des responsabilités

Principe d'ouverture/fermeture (OCP)

Inversion des dépendances (DIP)

Substitution de Liskov (LSP)

Séparation des interfaces (ISP)

Les design patterns et anti patterns

### **DEFINIR ET IMPLEMENTER LES PRINCIPAUX PATTERNS (1/3)**

---

Design patterns du gof et autres patterns

Les Design Patterns de comportement (itérateur, stratégie, Template Method, état, observateur, médiateur, visiteur)

### **DEFINIR ET IMPLEMENTER LES PRINCIPAUX PATTERNS (2/3)**

---

Design patterns du gof et autres patterns

Les design patterns de création (singleton, fabrique abstraite, Builder)

### **DEFINIR ET IMPLEMENTER LES PRINCIPAUX PATTERNS (3/3)**

---

Design patterns du gof et autres patterns

Les design patterns de structure (composite, pont, adaptateur, décorateur, façade, proxy, extension object, value object)

### **APPREHENDER LA DEMARCHE D'UTILISATION DES MODELES POUR APPLIQUER LE BON PATRON (1/4)**

---

Les design patterns de comportement

Identification des coopérations évolutives

Utilisation du design pattern stratégie

### **APPREHENDER LA DEMARCHE D'UTILISATION DES MODELES POUR APPLIQUER LE BON PATRON (2/4)**

---

Les design pattern de création

Rendre un système indépendant de la façon dont les objets sont créés

Utilisation des design patterns Factory (fabrique) et/ou Builder

## **APPREHENDER LA DEMARCHE D'UTILISATION DES MODELES POUR APPLIQUER LE BON PATRON (3/4)**

---

Les design patterns de création

Obtenir des objets "augmentés" (obtenir de nouvelles fonctionnalités pour des objets)

Identifier des structures de classes abstraites et évolutives

Utilisation des design patterns décorateur, composite

## **APPREHENDER LA DEMARCHE D'UTILISATION DES MODELES POUR APPLIQUER LE BON PATRON (4/4)**

---

Les design patterns de structure

Coopérations découplées, isolation des logiques ou dépendances internes

Utilisation des design patterns proxy et/ou façade

## **APPLIQUER LES DESIGN PATTERNS DANS UNE ARCHITECTURE TECHNIQUE DE CONCEPTION (1/2)**

---

Patterns et architecture

Couches logicielles

Approche par composants

Rappels du modèle observer

Modèle vue contrôleur (MVC)

## **APPLIQUER LES DESIGN PATTERNS DANS UNE ARCHITECTURE TECHNIQUE DE CONCEPTION (2/2)**

---

Patterns et architecture

Couches logicielles

Approche par composants

Couche d'accès aux données (DAO)

Framework et design patterns (DAO, singleton, injection de dépendances)

---