

CONCEPTION TECHNIQUE AVEC LES DESIGN PATTERNS

Enjeux

La conception orientée objet a longtemps été un art plutôt qu'une science.

Selon la définition de son inventeur, un Design Pattern décrit un problème récurrent dans un environnement donné puis décrit une solution à ce problème de telle manière que cette solution soit réutilisable à chaque fois qu'on rencontre le problème sans cependant qu'elle soit deux fois exactement la même.

Les Design Patterns fournissent ainsi des moyens de transmettre le savoir-faire et l'expérience dans le domaine de la conception orientée objet, permettant ainsi de l'industrialiser autant que possible.

Stratégie et techniques de conception

- La conception objet est une tâche complexe
- Raffinements successifs du modèle d'analyse

Design Patterns : définition

- Des solutions de conception « sur étagère »
- Définir le besoin technique : le client
- Définir les classes « types » du pattern
- Définir les collaborations entre ces classes
- Les CRC cards

Modèle d'analyse et architecture

- Définir les règles de conception applicables
- Définir les modèles techniques abstraits
- Systématiser les principes d'architecture
- Automatiser les principes d'architecture

Design Patterns : utilisation

- Description formelle d'un pattern
- Comment choisir un pattern ?
- Comment utiliser un pattern ?

Patterns de construction d'objets

- Rendre un système indépendant de la façon dont les objets sont créés
- Patterns étudiés : Abstract factory, Singleton, Prototype

Patterns de structuration

- Composer des objets pour obtenir de nouvelles fonctionnalités
- Identifier des structures de classes abstraites et évolutives
- Patterns étudiés : Adapter, Composite

Patterns de gestion des comportements

- Identifier des coopérations évolutives et découplées
- Patterns étudiés : Observer et MVC, Command Processor, Strategy, Template method

Patterns de communication

- Identifier des coopérations découplées utilisables dans un cadre distribué
- Patterns étudiés : Publisher-Subscriber, Proxy

Étude de cas Java

- Création d'une application Java en utilisant les patterns Observer, Singleton, Command Processor et Composite
- Éléments pour la personnalisation du module
- Démonstration de l'automatisation des patterns avec OBJECTEERING

Conclusion

- Formalisation des solutions techniques
- Notions et vocabulaire communs entre développeurs
- Réutilisation d'expertise technique

OBJECTIFS

- Comprendre les Design Patterns
- Savoir les mettre en œuvre dans une architecture technique de conception.

PUBLICS

Ces cours sont destinés aux chefs de projets, architectes et ingénieurs concepteurs.

Pré-requis :

- La connaissance d'UML est fortement recommandée.
- La connaissance des langages C++ ou Java est nécessaire.

Rapport Théorie/pratique :

- 60/40

RÉF : CDP 2 JOURS

16 - 17	janvier
13 - 14	mars
26 - 27	mai
28 - 29	juillet
11 - 12	septembre
17 - 18	novembre

770 €ht

Prix par personne.

