



Architecture orientée objet avec UML

Durée : Trois jours
Niveau : Intermédiaire

DESCRIPTION

L'objectif du cours est d'apprendre aux étudiants à appliquer un processus itératif, basé sur les cas d'utilisation, au développement de l'architecture d'un système logiciel. Le cours traite des considérations architecturales qui influencent l'organisation et les mécanismes clés du système et de leur présentation sous forme de vues architecturales à l'aide de la notation UML.

OBJECTIFS

À la fin du cours, les participants seront capables de:

- Identifier les acteurs d'un système et leurs objectifs
 - Définir la portée d'un système
 - Lire et interpréter les artefacts qui spécifient les exigences, notamment un modèle des cas d'utilisation
 - Énoncer et décrire les principes de base de l'orientation objet
 - Expliquer la relation entre une classe et un objet
 - Appliquer les concepts d'abstraction, d'encapsulation, d'héritage et de polymorphisme.
 - Comprendre les artefacts qui supportent l'architecture et leur impact sur le modèle de conception produit.
 - Identifier les abstractions clés à partir des cas d'utilisation, les modéliser sous forme de classes et leur attribuer des responsabilités
 - Définir l'organisation et les mécanismes clés d'un système
 - Présenter les artefacts architecturaux sous forme de vues à l'aide de la notation UML
-

CONTENU

- Concepts de base
 - Concepts de base de l'orientation objet
 - Introduction à la gestion des exigences avec les cas d'utilisation
 - Introduction à l'analyse et à la conception orientée objet
 - Définition d'une architecture candidate
 - Définition de l'organisation des sous-systèmes
 - Identification des abstractions clés
 - Identification des mécanismes d'analyse
 - Identification des réalisations de cas d'utilisation
 - Analyse du comportement
 - Identification des classes à partir du comportement des cas d'utilisation
 - Utilisation des stéréotypes d'analyse de la notation UML
 - Modélisation des aspects dynamique et statique d'une réalisation de cas d'utilisation
 - Conception détaillée de l'architecture
 - Identification des éléments de conception
 - Identification des mécanismes de conception
 - Description de la concurrence
 - Description de la distribution
 - Persistance
-

AUDIENCE

- Architectes de logiciels
 - Développeurs de logiciels
 - Spécialistes d'application
-

PRÉALABLES

- Bonne connaissance du cycle de développement d'une application logicielle