

## Enterprise Architect

### Modélisation UML 2.0 avec Enterprise Architect

En 10 ans, UML s'est imposé comme le standard en matière de langage de modélisation des systèmes informatiques. De l'expression des besoins au déploiement en passant par la conception, utiliser ou tout simplement comprendre UML est un incontournable de tout acteur du S.I., qu'il soit maîtrise d'ouvrage ou maîtrise d'œuvre.

Le logiciel Enterprise Architect de Sparx System permet de réaliser tous les diagrammes UML 2. Bien plus qu'un simple modèleur UML, il permet de mettre en œuvre toutes les phases et disciplines d'un projet.

Cette formation permet de comprendre UML à travers la maîtrise de cet outil.

#### Détails

- Code : MM-UMEA
- Durée : 3 jours ( 21 heures )

#### Public

- Analystes
- Architectes
- Chefs de projets
- Ingénieurs

#### Pré-requis

- Maîtrise des principes de programmation orientée objet

#### Objectifs

- Maîtriser les 13 diagrammes UML 2 et savoir les utiliser avec Enterprise Architect
- Savoir utiliser les bons diagrammes en fonction des phases projet
- Savoir utiliser les outils complémentaires d'Enterprise Architect

#### Programme

##### Présentation des architectures

- Architectures à objets distribués, architectures n-tiers
- Architectures Java EE et .NET
- Architectures SOA et Services Web
- Le MVC (Model View Controller) dans Enterprise Architect

- Organisation du système en modules
- Organisation des cas d'utilisation
- Spécification des cas d'utilisation
  - Modèle de documentation
  - Diagrammes d'activité

##### Présentation du Processus Unifié (UP)

- Historique et Objectifs
- Développement en cascade vs itératif
- Pratiques UP
- Disciplines UP
- Phases UP
- Créer un projet Enterprise Architect en utilisant UP

##### Analyse et conception UML avec Enterprise Architect

- Objectifs
- Analyse métier
- Entités métier
  - Associations
  - Multiplicité
  - Navigation
  - Rôles
- Utiliser les templates Enterprise Architect
- Conception
  - Modèles de conception
  - Diagramme de classe
  - Abstraction et héritage
  - Diagramme de classe
  - Interface et opérations
  - Diagramme de package
  - Comportement du système
  - Diagramme de séquence
  - Utiliser la bibliothèque Enterprise Architect de Design Patterns
- Génération de code avec Enterprise Architect

##### Modéliser les exigences avec Enterprise Architect

- Gérer les dictionnaires
- Modéliser les exigences
- Gérer les règles métier
- Périmètre du système
- Définition : acteurs
- Identification des exigences
  - Cas d'utilisation
  - Cas d'utilisation
  - Relations entre cas d'utilisation
  - Diagrammes de cas d'utilisation
- Gérer la traçabilité des exigences
- Analyse des exigences

- Générer les modèles à partir du code (reverse engineering)
- Génération de documentation avec Enterprise Architect

### UML2.0 : Modèle et méta-modèle

- Objectifs
- Extensions d'UML

- Gérer les stéréotypes
- Gérer les tagged value
- Définition d'un Profile
- Le méta modèle UML
- Le langage normalisé OCL (Object Constraint Language)

### Modalités

- **Type d'action** :Acquisition des connaissances
- **Moyens de la formation** :Formation présentielle - 1 poste par stagiaire - 1 vidéo projecteur - Support de cours fourni à chaque stagiaire
- **Modalités pédagogiques** :Exposés - Cas pratiques - Synthèse
- **Validation** :Exercices de validation - Attestation de stages