

Formation Analyse et conception avec UML 2.0



Comment passer des besoins des utilisateurs au code de l'application ? Autrement dit : « Comment obtenir le plus efficacement possible un code informatique opérationnel, complet, testé, et qui réponde le mieux possible au besoin des futurs utilisateurs ? ». Du cahier des charges au code, cette formation présente les meilleures pratiques de modélisation avec UML 2 sous la forme d'une étude de cas complète

Objectifs

- Savoir créer des modèles à l'aide d'UML 2.0 et des cas d'utilisation complets et efficaces
- Être capable de créer des diagrammes illustrant les interactions entre les objets
- Comprendre comment traduire les modèles de conception dans le langage objet souhaité

Public concerné

- Chefs de projet, analystes, concepteurs et développeurs

Pré requis

- La pratique d'un langage de programmation orienté objet est fortement conseillée

Une formation de 4 jours

Caractéristiques	Paris	Lyon
Tarif : 1950 € HT par personne	14/03/2011	04/04/2011
Numéro de formateur : 11753687675	16/05/2011	03/10/2011
Nombre d'heures : 28	22/08/2011	
Référence : OB210	07/11/2011	
Contact : Patrick LE GOFF		
Telephone : 01.76.60.66.10		
Email : contact@kaptive.com		

Description des modules

num	Module
1	Introduction
Détails	<ul style="list-style-type: none"> - La modélisation agile - Les bases d'UML 2 - La démarche proposée - Les études de cas, les outils
2	Cas d'utilisation
Détails	<ul style="list-style-type: none"> - Les bases : acteur, cas d'utilisation - Le diagramme de cas d'utilisation - La description textuelle des UC - Les bases du diagramme de séquence - Le diagramme de séquence système
3	Modèle du domaine
Détails	<ul style="list-style-type: none"> - Les concepts du domaine - Les attributs et associations - Les rôles et multiplicités - L'agrégation et la composition - Le diagramme des classes
4	Conception Objet préliminaire
Détails	<ul style="list-style-type: none"> - De l'analyse à la conception - Séquence et communication - Les messages et lignes de vie - Les bonnes pratiques de conception - Les classes logicielles
5	De la conception au code
Détails	<ul style="list-style-type: none"> - La définition structurelle des classes - Les corps des méthodes - L'outillage disponible
6	Retour sur les cas d'utilisation
Détails	<ul style="list-style-type: none"> - Les relations entre cas d'utilisation - UC et IHM, UC et tests, UC et gestion de projet
7	Retour sur le modèle du domaine
Détails	<ul style="list-style-type: none"> - La généralisation / spécialisation - Quelques compléments - La notion de package - Les bases du diagramme d'états - Une démarche de création
8	Retour sur la conception
Détails	<ul style="list-style-type: none"> - Le polymorphisme - Les interfaces, les Design Patterns - Le diagramme de composants, de déploiement
9	Compléments
Détails	<ul style="list-style-type: none"> - La modélisation métier - Les bases du diagramme d'activité
10	Conclusion
Détails	<ul style="list-style-type: none"> - Récapitulatif de la démarche

- Qu'est-ce qu'un modèle agile ?
- Ressources complémentaires