

# 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 : <a href="mailto:contact@kaptive.com">contact@kaptive.com</a>		

## Description des modules

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

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