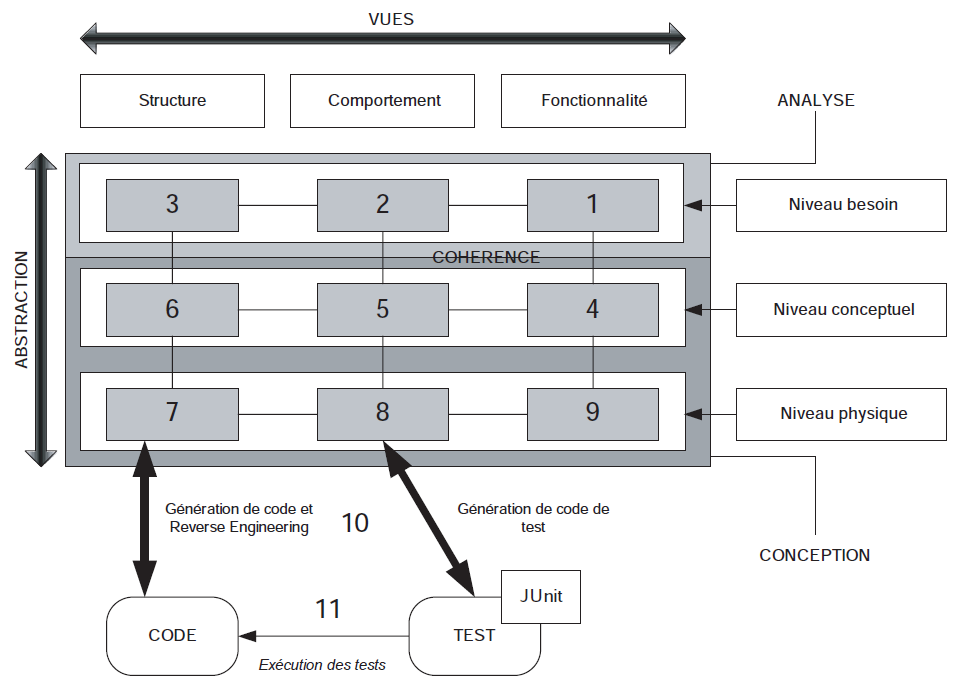
## UML Pour développeurs



Niveau besoin : correspond au recueil du besoin (cahier des charges + UML exprimant le besoin : use case et diagramme de séquence)

Niveau conceptuel : Diagrammes UML représentant la solution, et correspondant aux classes java développés

Niveau physique : Code dans la solution technique choisi.

Fonctionnalité : Diagramme de cas d’utilisation ( use case)  
Comportement : Diagramme de séquence   
Structure : Diagramme de classe.

Le niveau besoin et conceptuel ne sont pas reliés, mais il est possible de mettre des notes dans le conceptuel pour identifier les éléments correspondants au besoin résolus par ceux-ci.

Le niveau conceptuel et technique sont interconnecté en définissant des règles permettant aux outils UML de faire le reverse engineering et la génération de code. Cela permet au code de rester à jour sur le modèle, et vice versa. Le gain est de pouvoir voir un niveau au dessus pour appliquer des design pattern à l’uml, générer les tests Junit depuis le modèle, la documentation, et la correction des dépendances (éviter les dépendances croisés entre classes)

L’ordre de réalisation d’un projet partant de 0 va des étapes 1 à 9, à partir du cahier des charges.

Exemple de diagramme :

Diagramme de cas d’utilisation du système

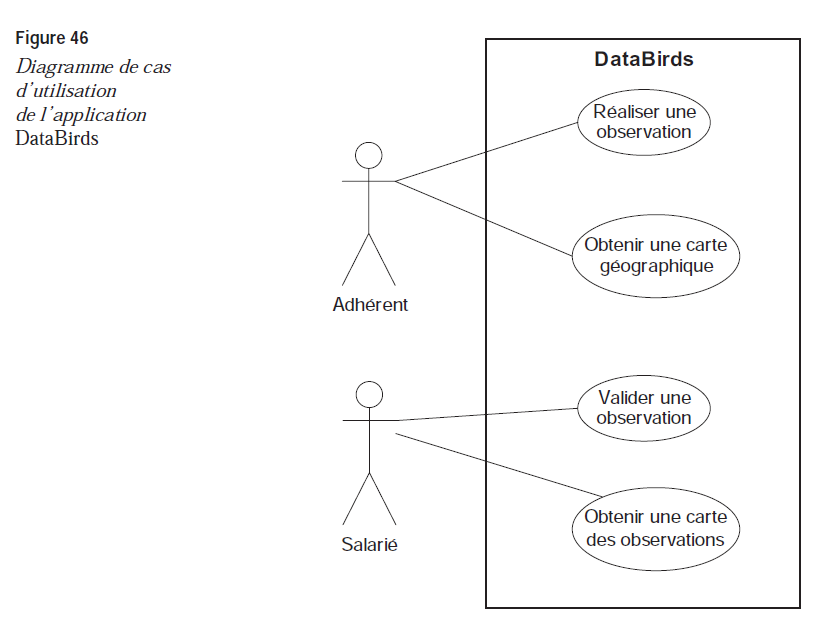


Diagramme de cas d’utilisation d’un composant du système

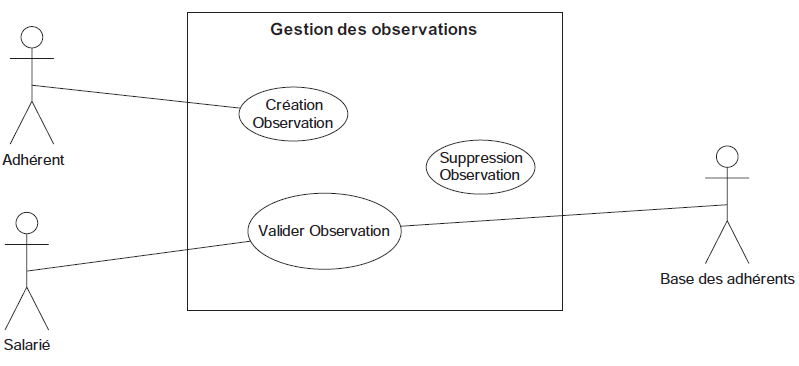


Diagramme de séquence d’un cas d’utilisation du composant

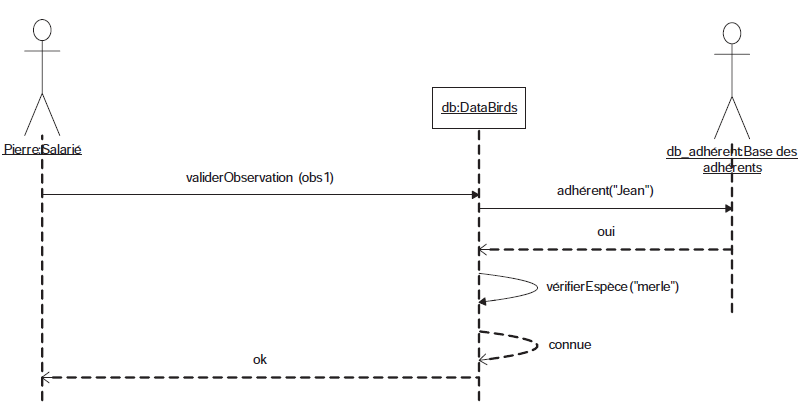


Diagramme de classe

