# Développement et Ingénierie des logiciels, exigence et modélisation par UML

## 1. Développement de logiciel

Le développement de logiciel consiste à étudier, concevoir, construire, transformer, mettre au point, maintenir et améliorer des logiciels. Un logiciel est créé progressivement par une équipe d'ingénieurs conformément à un cahier des charges établi par un client demandeur ou une équipe interne. Le logiciel est décomposé en différents modules et un chef de projet, ou architecte, se charge de la cohérence de l'ensemble.

## 2. Ingénierie logicielle

En ingénierie logicielle, une exigence peut être la description de ce qu'un système doit faire. Ces exigences se classent généralement en trois catégories :

1) Exigences fonctionnelles : Elles décrivent les caractéristiques du système ou des processus que le système doit exécuter. On trouve dans cette catégorie les règles métier.

2) Exigences non fonctionnelles : Elles décrivent les propriétés que le système doit avoir, comme les exigences techniques de sécurité informatique (confidentialité, intégrité, disponibilité), de performance, d'accessibilité.

3) Contraintes : Les limites du développement, comme définir un système d'exploitation spécifique ou un langage de programmation imposé.

## 3. UML (Unified Modeling Language)

UML est un langage standard conçu pour l'écriture des plans d'élaboration de logiciels. Il est utilisé pour visualiser, spécifier et documenter les systèmes logiciels. UML comprend 13 diagrammes principaux :

- Diagrammes structurels :  
 - Diagramme de classes  
 - Diagramme d'objets  
 - Diagramme de composants  
 - Diagramme de déploiement

- Diagrammes comportementaux :  
 - Diagramme de cas d'utilisation  
 - Diagramme de séquence  
 - Diagramme d'états-transitions  
 - Diagramme d'activités

## 4. Éléments fondamentaux d'UML

UML repose sur quatre éléments fondamentaux :

1) Les éléments structurels : Classes, interfaces, collaborations, cas d'utilisation, composants et nœuds.

2) Les éléments comportementaux : Interactions, machines à états.

3) Les éléments de regroupement : Paquetages.

4) Les éléments d'annotation : Notes.

## 5. Relations dans UML

UML utilise quatre types de relations principales :

1) Dépendance : Relation sémantique entre deux éléments.

2) Association : Relation structurelle entre classes.

3) Généralisation : Relation de spécialisation/héritage.

4) Réalisation : Relation entre interfaces et classes.

La modélisation UML a permis de formaliser les exigences fonctionnelles et non fonctionnelles du système, de concevoir une architecture robuste et de faciliter la communication entre les différentes parties prenantes du projet ExtremeSharing.org. Cette approche méthodique a assuré la cohérence entre les besoins exprimés et la solution implémentée.

**Référence : Grady Booch, James Rumbaugh, Ivar Jacobson, « Le guide de l'utilisateur UML », édition Eyrolles, 2001.**