Introduction à l'UML (Unified Modeling Language)

Modéliser les logiciels pour mieux les comprendre

Réalisée par:
Ben Béchir Chaima

Qu'est-ce que l'UML ?

Un langage de modélisation visuelle

Qu'est-ce que l'UML ?

- **Définition :** L'UML (Unified Modeling Language) est un langage graphique et standardisé, utilisé pour la conception et la modélisation de systèmes logiciels. Il fournit un ensemble de symboles et de règles pour créer des schémas visuels.
- **Objectif :** Faciliter la compréhension d'un système complexe entre les différentes parties prenantes : développeurs, architectes, chefs de projet et clients.
- **Message clé :** L'UML n'est pas un langage de programmation. C'est un langage de **conception et de communication**. On ne code pas en UML, on dessine le plan avant de construire le logiciel.

Les deux types de diagrammes UML

Structure vs Comportement

1. Les diagrammes de structure :

- ✓ Décrivent l'architecture statique du système, ses composants et les relations entre eux.
- ✓ Répondent à la question : "De quoi le système est-il fait ? "
- ✓ **Exemples :** Diagramme de classes, diagramme de composants.

2. Les diagrammes de comportement :

- ✓ Décrivent la manière dont le système se comporte, ses interactions et ses processus dynamiques.
- ✓ Répondent à la question : "Comment le système fonctionne-t-il ? "
- ✓ **Exemples :** Diagramme de cas d'utilisation, diagramme de séquence.

Diagramme de cas d'utilisation

Comprendre les besoins du système

- **Rôle :** Décrire les fonctionnalités du système du point de vue de l'utilisateur. Il est souvent le premier diagramme réalisé.

- **Composants :**
  - ✓ **Acteur :** Une personne, une organisation ou un autre système qui interagit avec le système. Symbolisé par un bonhomme.
  - ✓ **Cas d'utilisation :** Une fonction du système qui apporte une valeur à l'acteur. Symbolisé par une ellipse.
  - ✓ **Lien :** Une ligne reliant un acteur à un cas d'utilisation, montrant une interaction.

- **Exemple :** Un utilisateur (**acteur**) peut "Passer une commande" (**cas d'utilisation**).

Diagramme de classes

Modéliser la structure des données

- **Rôle :** Représenter la structure statique du système en termes de classes, d'attributs et de méthodes. C'est le plan de base pour la programmation orientée objet.

- **Composants :**
  - ✓ **Classe :** Représentée par un rectangle avec trois sections (nom, attributs, méthodes).
  - ✓ **Attributs :** Les propriétés de la classe (ex: **Personne** a un attribut **nom**).
  - ✓ **Méthodes :** Les actions que la classe peut exécuter (ex: **Personne** peut **marcher()**).
  - ✓ **Relations :** Associations, agrégation, composition, héritage.

- **Exemple :** Une classe **Client** liée à une classe **Commande**.

Diagramme de séquence

Visualiser les interactions dans le temps

- **Rôle :** Montrer comment les objets interagissent les uns avec les autres pour accomplir une tâche, en suivant une séquence chronologique.

- **Composants :**
  - ✓ **Ligne de vie (Lifeline) :** Représente un objet participant à l'interaction.
  - ✓ **Message :** Une flèche qui indique une communication entre les objets.
  - ✓ **Barre d'activation :** Un rectangle qui montre la période pendant laquelle un objet est actif.

- **Exemple :** Un **utilisateur** envoie un message à un **système** pour se connecter.

Diagramme d'activité

Modéliser les flux de travail

- **Rôle :** Représenter un processus métier ou un algorithme en montrant le flux de contrôle d'une activité à l'autre.

- **Composants :**
  - ✓ **Nœud d'activité :** Une action ou une étape du processus.
  - ✓ **Flèche :** Le flux de contrôle (le passage d'une activité à l'autre).
  - ✓ **Point de décision :** Un losange pour représenter une condition ou un choix.
  - ✓ **Point de départ/fin :** Un cercle noir pour le début et un cercle encerclé pour la fin.

- **Exemple :** Un diagramme pour le processus de "commande d'un produit en ligne".

Les avantages de l'UML

Pourquoi utiliser l'UML ?

- **Clarté et compréhension :** L'UML rend les systèmes complexes plus faciles à comprendre et à visualiser.
- **Meilleure communication :** Il sert de langage commun entre les équipes techniques et non techniques.
- **Réduction des erreurs :** La modélisation permet de détecter les incohérences et les erreurs de conception avant le codage.
- **Documentation :** Les diagrammes servent de documentation claire et standardisée du système.

Conclusion et résumé

En bref

Résumé :

- L'UML est un langage graphique pour modéliser des systèmes.
- Il se divise en diagrammes de structure (quoi) et de comportement (comment).
- Les diagrammes de cas d'utilisation et de classes sont les plus courants pour la conception initiale.
- L'UML n'est pas une obligation, mais un outil puissant pour les projets complexes, favorisant la clarté et la collaboration.

Merci de votre attention!