



Techniques complémentaires de Production de Logiciels

Responsable : Dr. DIAW

Formation : DSTI

Niveau : 2

Semestre: 2



Introduction

- UML n'est pas une méthode
- Comment passer des besoins des utilisateurs au code exécutable
- UML propose diverses notations pour décrire les différentes facettes d'un système matériel ou logiciel
- UML n'explique pas les différentes étapes pour passer des besoins au code du logiciel



Méthode: Démarche et formalisme

- Démarche : succession d'étapes pour
 - mieux maîtriser le déroulement d'un projet
 - Meilleure visibilité pour les utilisateurs sur certains résultats intermédiaires et garantir que le résultat final sera celui attendu
- Formalisme défini par:
 - Un langage formel
 - Un langage semi-formel généralement graphique
 - Un langage naturel
- Fonction :
 - Représenter le monde réel tel qu'il est perçu par le concepteur
 - Outil de communication entre informaticiens et utilisateurs
 - Constitué par un ensemble de modèles permettant d'assurer une bonne compréhension des besoins des utilisateurs



Modèle

- Représentation abstraite de la réalité qui exclut certains détails du monde réel
- un artéfact formulé dans un langage de modélisation tel que UML, et décrivant un système à travers différents types de diagrammes [Rumbaugh et al., 1999].
- Dans [Bézivin et al., 2001], un modèle est défini comme une représentation d' un ou d' une partie d' un système construit pour un objectif précis. Le modèle doit répondre aux questions que les utilisateurs se posent sur le système qu' il représente.
- Dans [Kleppe et al., 2003] un modèle est défini comme une description d' un ou d' une partie d' un système dans un langage bien défini. Un langage bien défini est un langage qui a une syntaxe et une sémantique bien définie et qui est interprétable par un outil.



Cycle de vie

- La qualité du processus de fabrication est garante de la qualité du produit
- Pour obtenir un logiciel de qualité, il faut en maîtriser le processus d'élaboration
- La vie d'un logiciel est composée de différentes étapes La succession de ces étapes forme le cycle de vie du logiciel
- Il faut contrôler la succession de ces différentes étapes



Etude de faisabilité

- Déterminer si le développement proposé vaut la peine d'être mis en œuvre, compte tenu de attentes et de la difficultés de développement
- Etude de marché : Déterminer s'il existe un marché potentiel pour le produit.



Analyse et Spécification des besoins

- Déterminer les fonctionnalités que doit posséder le logiciel
- **Collecte des exigences** : obtenir de l'utilisateur ses exigences pour le logiciel
- **Analyse du domaine** : déterminer les tâches et les structures qui se répètent dans le problème



Organisation du projet

- Déterminer comment on va développer le logiciel
- **Analyse des coûts** : établir une estimation du prix du projet
- **Planification** : établir un calendrier de développement
- **Assurance qualité du logiciel** : déterminer les actions qui permettront de s'assurer de la qualité du produit fini
- **Répartition des tâches** : hiérarchiser les tâches et sous-tâches nécessaires au développement du logiciel



Conception

- Déterminer la façon dont le logiciel fournit les différentes fonctionnalités recherchées
- Conception générale
 - Conception architecturale : déterminer la structure du système
 - Conception des interfaces : déterminer la façon dont les différentes parties du système agissent entre elles
- Conception détaillée : déterminer les algorithmes pour les différentes parties du système



Implémentation

- Ecrire le logiciel



Test

- Essayer le logiciel sur des données d'exemple pour s'assurer qu'il fonctionne correctement
- **Tests unitaires** : faire tester les parties du logiciel par leurs développeurs
- **Tests d'intégration** : tester pendant l'intégration
- **Tests de validation** : pour acceptation par l'acheteur
- **Tests système** : tester dans un environnement proche de l'environnement de production
- **Tests Alpha** : faire tester par le client sur le site de développement
- **Tests Bêta** : faire tester par le client sur le site de production
- **Tests de régression** : enregistrer les résultats des tests et les comparer à ceux des anciennes versions pour vérifier si la nouvelle n'en a pas dégradée d'autres



Livraison

- Fournir au client une solution logicielle qui fonctionne correctement
- **Installation** : rendre le logiciel opérationnel sur le site du client
- **Formation** : enseigner aux utilisateurs à se servir du logiciel
- **Assistance** : répondre aux questions des utilisateurs



Maintenance

- Mettre à jour et améliorer le logiciel pour assurer sa pérennité
- Pour limiter le temps et les coûts de maintenance, il faut porter ses efforts sur les étapes antérieures