

INF2015 – Développement de logiciels dans un environnement Agile

Méthodes de développement et environnements de développement

Jacques Berger

Objectifs

Présenter le modèle itératif

Introduire l'écosystème de développement

Prérequis

Aucun

Modèles de développement

Modèle en cascade

Modèle itératif

Cascade

Approche conservatrice

Analyse des besoins

Analyse fonctionnelle

Analyse organique (conception)

Réalisation

Tests

Maintenance

Cascade

Idée de base : On analyse tout le système avant de passer à l'étape suivante

Reconnue pour les dépassements de coûts et les longs délais de livraison

Cascade

Fonctionne bien lorsque...

Toutes les spécifications sont connues dès le début du projet

Les spécifications ne changent pas

Ne fonctionne pas bien...

Dans la plupart des cas... les projets logiciels ont des spécifications instables

Cascade

Les changements coûtent chers et ne sont pas les bienvenues

Donne l'impression que le projet avance mais rien de concret n'est réalisé

Difficile d'avoir un logiciel de qualité lorsqu'on effectue l'assurance-qualité à la fin du projet

Sujet aux erreurs architecturales

Modèle itératif

Développement
Itératif
Incrémental
Évolutif

Modèle itératif

Itératif : un logiciel exécutable est prêt à la fin de chaque itération

Incrémental : des fonctionnalités sont ajoutés au fur et à mesure que le projet avance

Évolutif : le logiciel évolue constamment (qualité, performance, etc.)

Modèle itératif

Itération

Cycle de développement complet
(Besoins-fonctionnel-conception-code-tests)

De courte durée

Un sous ensemble des fonctionnalités

Raffinement continu

Modèle itératif

Gestion des risques

On commence par le plus risqué
(Complexité - architecture)

Preuves de concept

Prototypage

Modèle itératif

Description d'une itération

Le début : les besoins, retour sur l'itération précédente (tirer des leçons durant le projet)

Le développement : emphase sur la qualité du code

La fin : tests d'intégration, livrables et feedback

Modèle itératif

Mentalité «en cascade»

Combattre l'envie de tout analyser d'avance

Éviter la pollution d'une mentalité en cascade
dans un projet itératif

Attitude malsaine pour le projet

Documents

Les différents documents utilisés durant le projet :
Spécifications du logiciel
Diagrammes de classes, de séquences, etc.

Les diagrammes servent à la communication dans l'équipe et non à la documentation du logiciel

Documents

On génère les diagrammes à partir du code de l'itération précédente

On dessine les diagrammes en équipe sur un tableau et on prend une photo du diagramme

Méthodes

Les méthodes agiles mettent l'accent sur :

- La communication à l'intérieur de l'équipe
- La participation du client
- L'amélioration de la productivité (outils)
- Le raffinement continu du code (refactoring)

Livrables

Le code source

Les tests

Les fichiers connexes (WAR, JAR, SQL, ANT, etc)

Les versions du logiciel

Les dépendances et leurs versions

Projet construit (déployé)

Livrables

La documentation
Pour développeurs
Pour opérateurs
Pour gestionnaires
Pour utilisateurs
Wiki
Diagrammes
Etc.

Normes

Les normes en génie logiciel
Processus normalisés

Organismes principaux

IEEE

ISO

SEI

Exemple : Swebok

Normes

Les normes offrent une structure de base au projet :

Documents

Méthodes

Elles n'imposent pas de façon de faire mais suggèrent des pratiques qui ont faites leurs preuves.

On adapte les normes à chaque projet.

Normes

Les normes ont plutôt une approche basée sur la discipline

Comment trouver un équilibre entre la discipline et l'agilité?

Écosystème

Écosystème de développement : plusieurs outils
logiciels qui cohabitent ensemble
Le coffre d'outils du développeur

Écosystème

Ce que contient l'écosystème :

Éditeur de texte

IDE

Shell

Gestionnaire de sources

Outils de construction

Outils de qualité (lint, beautifier, tests)

Écosystème

Impact sur le développeur

Impact sur l'équipe de développement

Écosystème

Un seul IDE pour tous le monde ou chaque développeur peut utiliser l'IDE ou l'éditeur de son choix?

Impact sur :

Licenses

Support

Travail d'équipe

Écosystème

Contexte d'entreprise vs contexte open-source

Écosystème

Logiciel libre

Indépendance envers le système d'exploitation

Indépendance envers un IDE

Organisation des librairies

IDE vs Éditeur de texte + Shell