

Organisation et déroulement d'un projet logiciel

I – Projet logiciel : définition

1.1 Selon vous, qu'est-ce qu'un projet logiciel ? Quelles sont ses dimensions et ses contraintes ?

Un projet est une activité temporaire décidée en vue de produire un résultat unique (produit ou service). (Définition de l'Afnor – 1998).

Dimensions et contraintes : (dessin)

Il faut réaliser ce qui est demandé en respectant le budget et le planning prévu, avec un certain niveau de qualité requis.

1.2 Donner deux conditions nécessaires pour que le projet puisse être réalisé

1. Être capable de bien identifier les besoins du client et le cahier des charges.
2. Se donner des critères d'achèvement après chaque phase (tests et démarches de qualité).

1.3 Par contraste, quelles sont à votre avis les principales "maladies" d'un projet, ce qui conduit à un échec ?

- Manque de communication
- Erreurs de conception
- Tentation intellectuelle : on définit un projet trop abstrait
- Spécification rampantes : le cahier des charges n'en finit plus en fonctionnalités
- Mauvaise estimation des délais : cause de stress et de panique
- Le projet forcé : le chef de projet a mal choisi son équipe, pas assez motivée
- L'arthrose : blocage du projet car une tâche non aboutie bloque les autres
- Problème de coordination de l'équipe, problème de compétences
- Dégénérescence : trop grand écart entre ce que voulait le client et ce qui a été réalisé

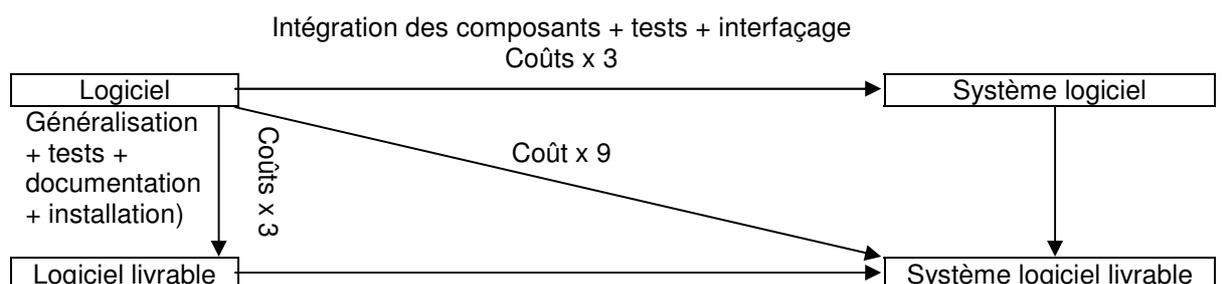
1.4 On s'intéresse maintenant à la typologie des projets logiciels. Quels sont les principaux facteurs qui les différencient ?

3 Facteurs :

- Nature de la livraison : logiciel, système complet, conseil, audit, service, étude...
- Taille : coût (budget), participants internes et externes, durée...
- Environnement social : organisation (entreprise, administration, recherche...), culture d'entreprise, problèmes politiques...

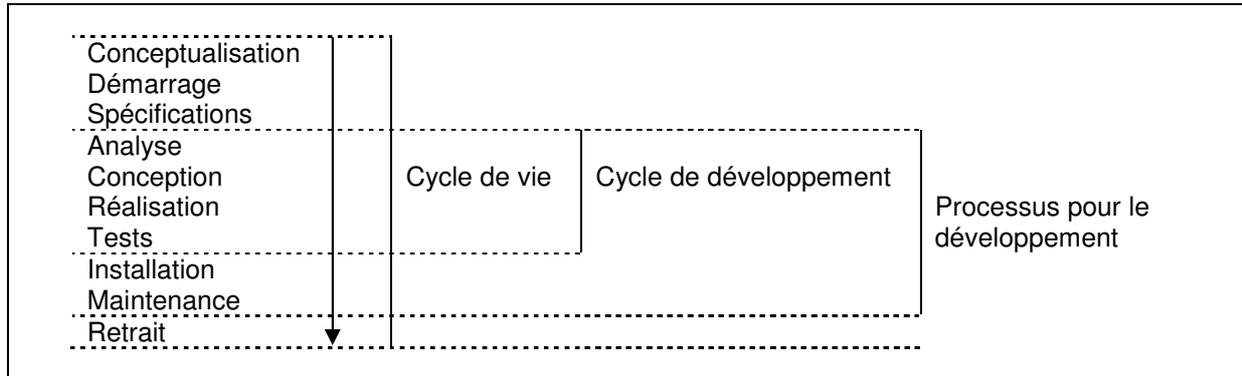
1.5 Qu'est-ce qu'un logiciel ? Comment passe-t-on du logiciel au système livrable final ?

Un logiciel est un ensemble de programmes et de manuels (cahier des charges, analyses, conception, manuel d'utilisateur, manuel du programmeur...)



II – Organisation et déroulement d'un projet logiciel

2.1 Cycle de vie d'un logiciel versus cycle de développement. Expliquer les différences.



2.2 Les processus du cycle de développement logiciel se divisent en trois catégories :

- processus de développement regroupant les activités contribuant directement à la réalisation des fournitures.
- processus transversaux regroupant des activités de même nature qui interviennent dans plusieurs phases du processus de développement.
- processus de support et de gestion regroupant les activités assurant le fonctionnement du projet.

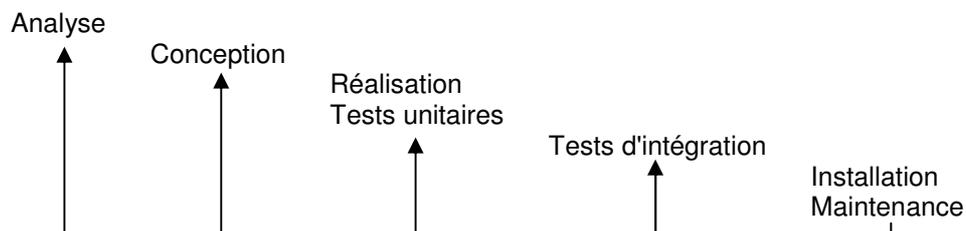
Ces trois composantes du cycle de développement du logiciel sont toutes essentielles à la conduite du développement du logiciel.

Lister les activités apparaissant dans le développement du logiciel et les rattacher à une classe de processus

- processus de développement : analyse, conception, réalisation, tests, installation, maintenance
- processus transversaux : planning, formation, revues, résolutions de problèmes, gestion de risques, gestion de la documentation, entretiens, enquêtes techniques, tests.
- Processus de support : gestion de projets, gestion de qualité, gestion de configuration

2.3 Rappeler les quatre phases principales d'un processus de développement. Montrer comment ces phases s'articulent lorsqu'on choisit un modèle en cascade, puis un modèle en processus évolutif.

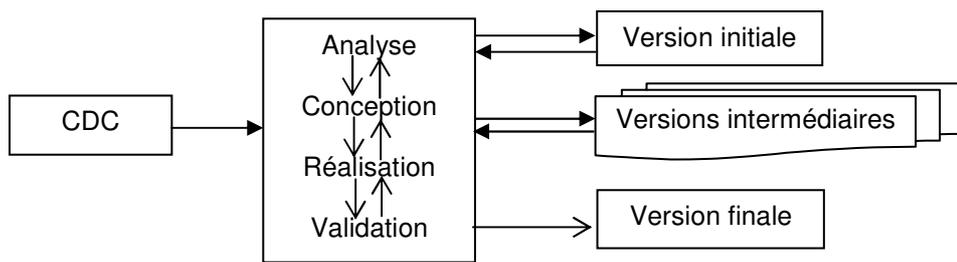
Processus en cascade :



Avantages : simple, phases bien identifiées avec des jalons

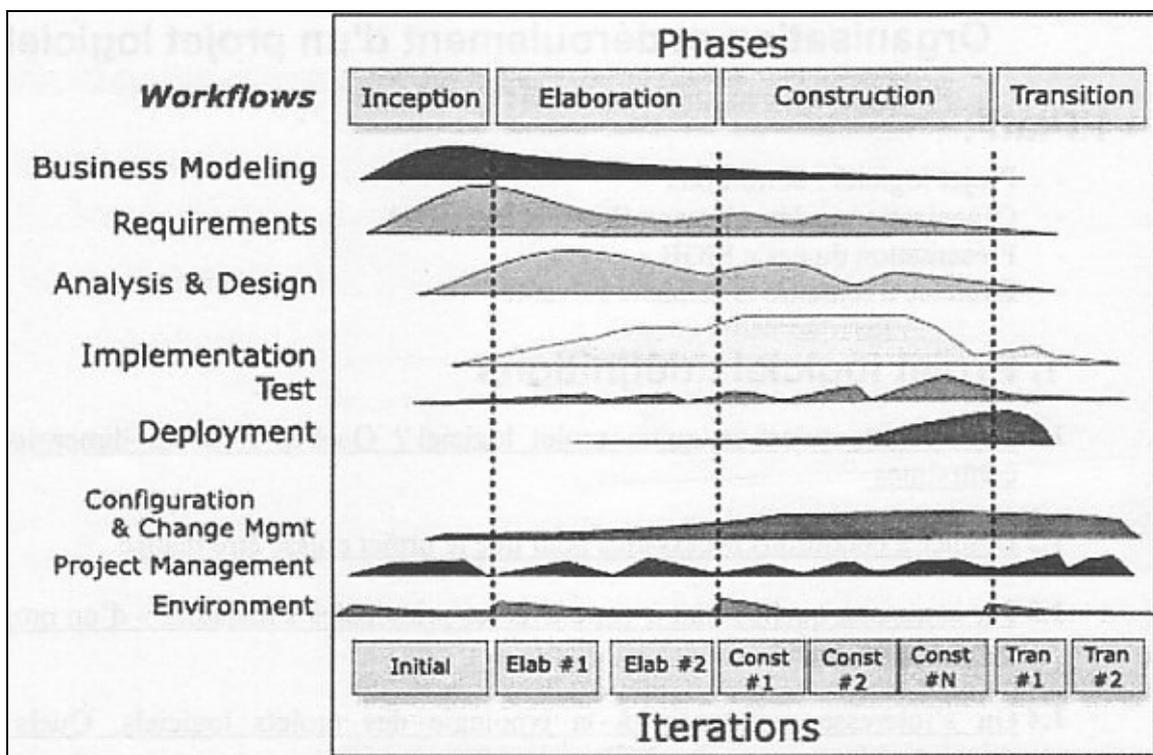
Inconvénients : tests trop tardifs pour revenir en arrière (coût !), structure pas très souple.

Processus évolutif :



Avantages : plus souple, plus proche de ce que veut le client
 Inconvénients : complexe, coûteux, on ne sait pas toujours où on en est.

2.4 On s'intéresse maintenant au processus unifié. La figure suivante le décrit en terme d'activités et cycle de vie :



Initialisation : idées, faisabilité, CDC, coûts, organisation, plan de développement
Elaboration : spécifications et analyse, diagramme de cas, architecture logicielle et matérielle, identification des risques (maladies).
Construction : conception, réalisation, tests unitaires, intégration, tests, gestion de versions.
Transitions : optimisation du code, version bêta (plus d'ajouts de fonctionnalité), test aux utilisateurs et correctifs, manuels d'utilisation, formation des utilisateurs, mise en place d'une maintenance.

2.5 Nous nous intéressons maintenant aux problèmes d'organisation du travail au sein de l'équipe d'un projet logiciel

2.5.a Tout d'abord, quelles sont selon vous les causes classiques de dysfonctionnement au sein d'une équipe ?

- Divergences d'intérêts
- Ambiguïté des rôles
- Rétention d'informations stratégiques
- Instabilité d'objectifs
- Batailles des chefs
- Conflits d'intérêts
- Chef de projet pas crédible
- Non engagement de la hiérarchie
- Mauvaise sélection du personnel
- Problèmes d'engagement personnel de l'équipe
- Problèmes de communication

2.5.b Afin notamment d'éviter ces dysfonctionnement quels principes de base doit suivre le chef de projet ?

- Eviter le culte de la personnalité
- Ne pas confondre moyens et objectifs
- Etre capable de gérer l'action : où on en est, les planning, aller voir les gens et les motiver, maîtriser le temps
- Savoir déléguer !

2.5.c Enfin, pour compléter cette présentation des organisation d'un projet logiciel, il faut noter que le développement est organisé en petites équipes, chaque équipe de développement pouvant éventuellement posséder un chef d'équipe chargé des relations avec le chef de projet. Une équipe commence avec deux personnes, mais le fonctionnement d'une équipe de deux personnes en entreprise est très différent de celle d'un binôme à l'université.

Afin de discuter ce point, comment décririez-vous vos expériences de travail en équipe, et notamment en binôme, aux cours de vos études ?

III – Présentation du cas SIGB