

Introduction aux méthodes Agiles

LP- IAM

Département Informatique- ESTS

Métier du chef de projet

Enrichissant

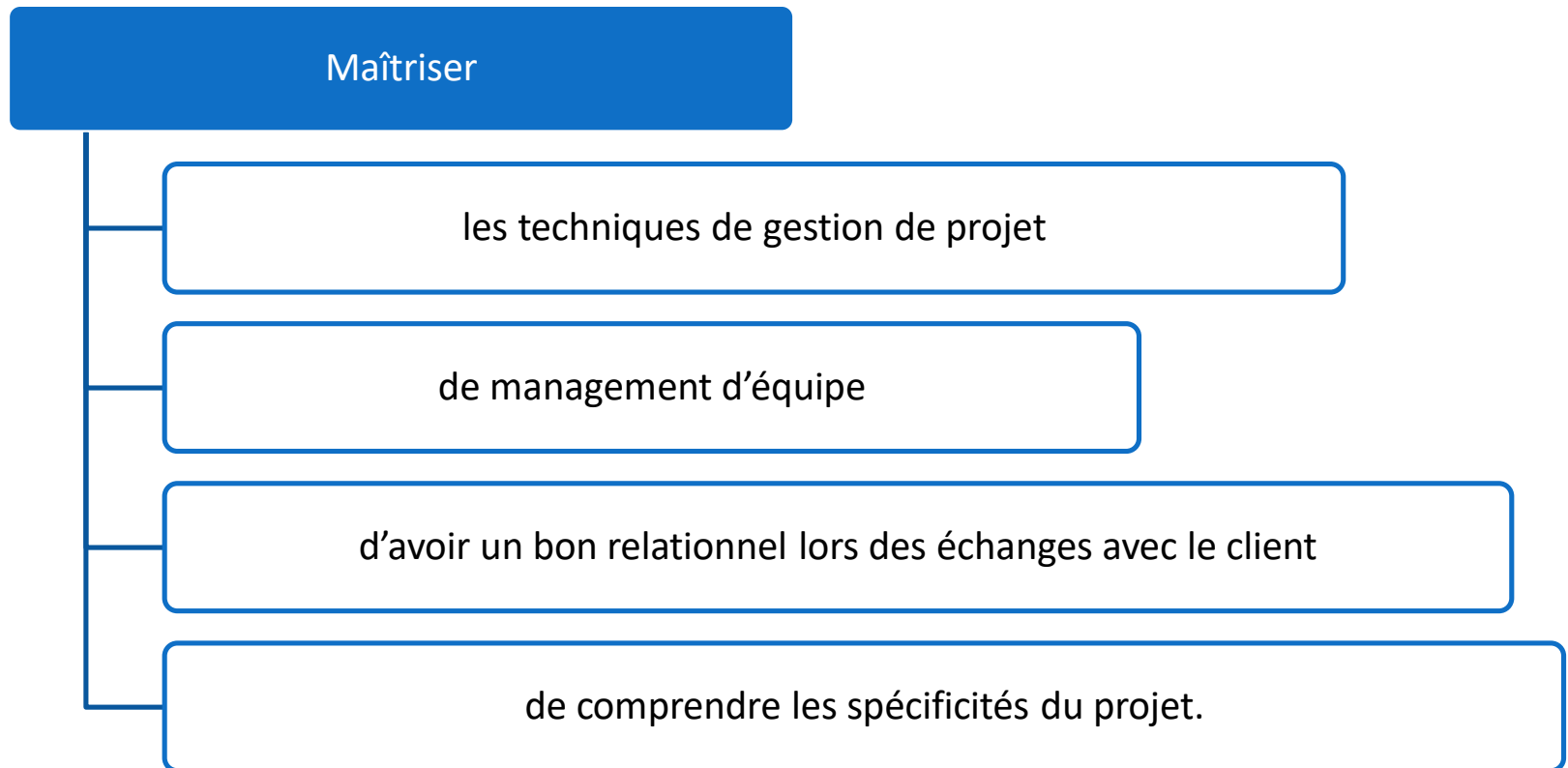
- Maîtrise plusieurs domaines

Complexe

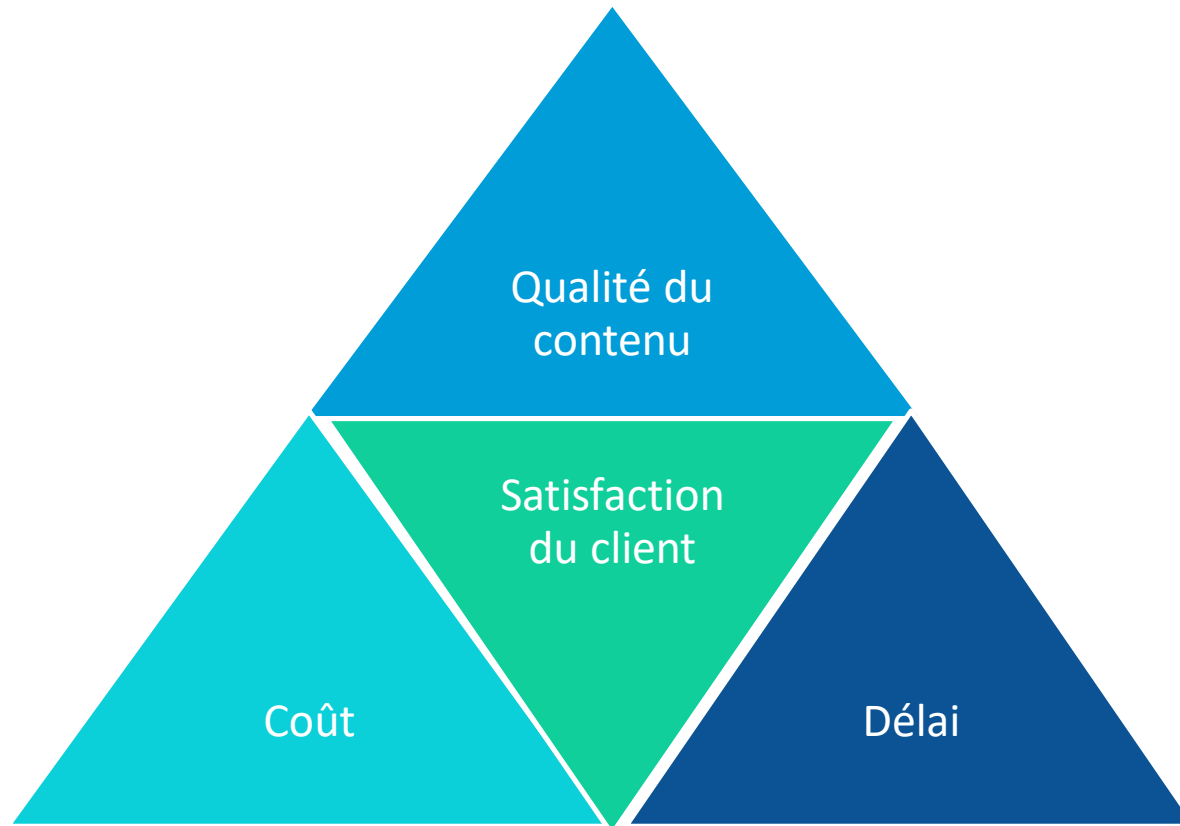
- Évoluer dans des environnements différents
- Qui sont en constantes évolutions

Le métier du chef de projet

Le chef de projet doit donc avoir plusieurs compétences:



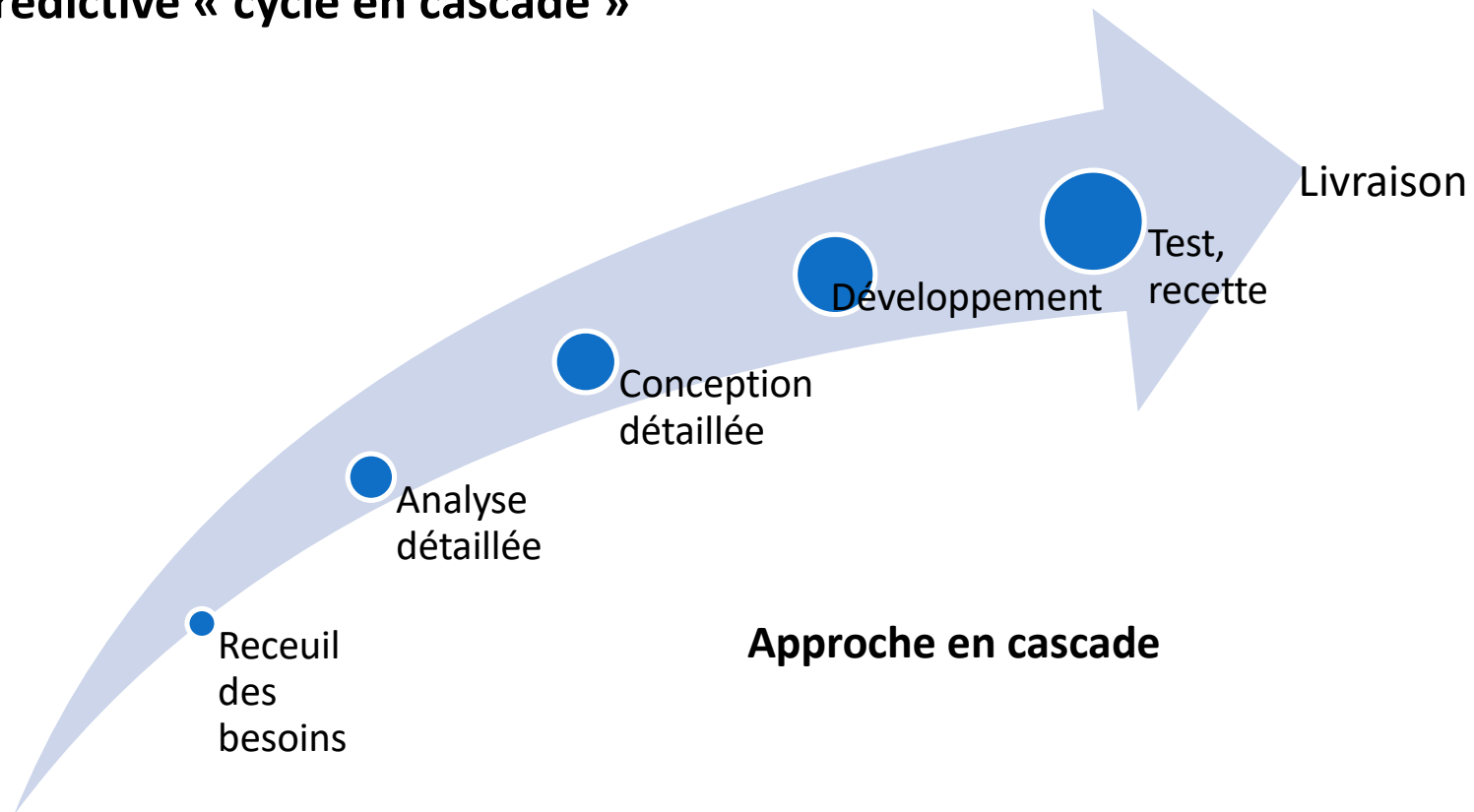
Objectif du chef de projet



Pour atteindre son objectif, le chef de projet peut avoir recours à plusieurs méthodes de gestion de projet.

Les méthodes classiques

Depuis toujours, les projets sont gérés avec la méthode dite « classique » qui se caractérise par recueillir les besoins, définir le produit, le développer et le tester avant de le livrer. On parle alors ici d'une approche **prédictive** « **cycle en cascade** »



Les méthodes classiques

- Comme son nom l'indique, il s'agit ici de prévoir des phases séquentielles où il faut **valider l'étape précédente pour passer à la suivante**.
- Le chef de projet doit alors s'engager sur **un planning précis** de réalisation du projet en prévoyant des jalons de débuts et fins de phases ainsi que les tâches à effectuer.
- **Il faut tout faire correctement du premier coup** car cette méthode ne peut pas permettre de retours en arrière.
- Une décision ou un problème rencontré dans une phase peuvent remettre en cause partiellement ou totalement les phases précédentes validées.

Les méthodes classiques

- Dans un cycle « en cascade » **les risques sont détectés tardivement** puisqu'il faut attendre la fin du développement pour effectuer la phase de test.
- Plus le projet avance, plus l'impact des risques augmente : **il sera toujours plus difficile et coûteux de revenir en arrière lorsqu'on découvre une anomalie tardivement.**
- Afin d'anticiper au mieux ces risques il est nécessaire de produire **des documents très détaillés en amont** (recueil des besoins, cahier des charges, les maquettes, etc...) qui seront validés par le client.
- Néanmoins, ces documents restent théoriques et conceptuels jusqu'à ce que le dispositif soit testé dans des conditions réelles ; le client validera le contenu papier (conception, maquette, développement fonctionnalités etc...) mais sera toujours plus sensible à ce qu'il verra sur son écran.

L'approche traditionnelle de gestion de projet

- Une approche dite « **traditionnelle** » attend généralement du client une expression détaillée et validée du besoin en entrée de réalisation, laissant peu de place au **changement**.
- **La réalisation dure le temps** qu'il faut et le rendez vous est repris avec le client pour la recette.
- Cet **effet tunnel peut être très** préjudiciable et conflictuel, on constate souvent un déphasage entre le besoin initial et l'application réalisée. On se rapporte alors aux **spécifications validées et au contrat**.
- De plus il n'est pas rare que certaines fonctionnalités demandées se révèlent finalement inutiles à l'usage alors que d'autres, découvertes en cours de route, auraient pu donner plus de valeur au produit.

Causes d'échecs de l'approche traditionnelle

Le «Standish Group» a effectué des enquêtes et a fait le constat suivant:

Une enquête de 1994 :

- 31 % des projets informatiques sont arrêtés en cours de route,
- 52 % n'aboutissent qu'au prix d'un important dépassement des délais et du budget tout en offrant moins de fonctionnalités qu'il n'en était demandé;
- seuls 16 % des projets peuvent être considérés comme des succès.

Une enquête renouvelée en 2008

- indique un taux de réussite de 35%, ce qui est plutôt positif mais demeure très faible. Le problème reste entier.
- Parmi les motifs d'échecs, arrivent en tête :
 - Manque d'implication des utilisateurs finaux : 12,8 %.
 - Changement de spécifications en cours de projet : 11,8 %.

Quels en sont les motifs d'échec, selon les professionnels de la gestion de projet ?



- 37% le manque d'objectifs et/ou d'étapes clairement définis et réalisables pour mesurer les progrès ;
- 19% une mauvaise communication ;
- 18% le manque de communication avec la direction ;
- 14% la résistance des employés ;
- 9% un financement insuffisant.

Approche Agile plutôt que méthode Agile

- Le terme « méthode » est trop limitatif pour parler de cette façon de concevoir, développer et délivrer un logiciel. Il s'agit de bien plus qu'une méthode.
- On parle plutôt de **paradigme Agile, d'état d'esprit Agile, de philosophie Agile, de culture Agile ou encore d'approche agile, de mouvement Agile, de courant Agile, de pratiques agiles, etc.**
- Cependant ,on parle de « **méthodes agiles** » pour **définir les méthodes qui relèvent de ces pratiques.**

Une autre approche de gestion de projet

- Le terme « Agile » définit une approche de **gestion de projet qui est à l'opposé des approches** traditionnelles prédictives et séquentielles de type **cycle en V ou en cascade**
- **La notion** même de « **gestion de projet** » est remise en question au profit de « **gestion de produit** ».
 - On raisonne davantage « **produit** » que « **projet** ».
 - L'objectif d'un projet est de donner naissance à un produit.

Approche Agile

L'approche Agile propose

- au contraire de réduire considérablement voire complètement cet **effet tunnel** en donnant davantage de visibilité, **en impliquant le client du début à la fin du projet**
- En adoptant un **processus de développement itératif et incrémental.**
- Elle considère que **le besoin ne peut être figé et propose au contraire de s'adapter aux changements de ce dernier.** Mais pas sans un minimum de règles.

Fonctionnement des méthodes agiles

- Les méthodes agiles partent du principe que spécifier et planifier dans les détails l'intégralité d'un produit avant de le développer (**approche prédictive**) est contre productif.

Fonctionnement des méthodes agiles

- Dans le cadre d'un projet de développement logiciel, le client
 - élabore sa vision du produit à réaliser
 - liste les fonctionnalités ou les exigences de ce dernier en fixant des priorités.
 - Il soumet cette liste à l'équipe de développement et communique directement avec elle.
- L'idée consiste à fixer un premier objectif du produit à court terme et se lancer dans sa réalisation. Une fois ce premier objectif atteint, on fait le point et on adapte la suite en fonction du résultat et des contraintes éventuelles. Et ainsi de suite jusqu'à la réalisation du produit final. On parle alors **d'une approche empirique**.

Fonctionnement des méthodes agiles

- En soumettant la liste des exigences à l'équipe de développement, le client est en contact direct avec elle. Celle-ci estime alors le coût de chaque élément de la liste. On peut ainsi se faire une idée approximative du budget global.

Fonctionnement des méthodes agiles

- L'équipe sélectionne ensuite une portion des exigences à réaliser dans une portion de temps courte appelée **itération**.
- Chaque itération inclut **des travaux de conception, de spécification fonctionnelle et technique quand c'est nécessaire, de développement et de test**.
- A la fin de chacune de ces itérations, le produit partiel mais utilisable est montré au client. Ce dernier se rend compte alors, très tôt, du travail réalisé et de l'alignement sur le besoin.
- L'utilisateur final peut donc se projeter dans l'usage du produit et émettre des feedbacks précieux pour les futures itérations. La visibilité ainsi offerte est essentielle.
- Cette transparence peut également apporter davantage de confiance et de collaboration dans la relation client/fournisseur.
- Les risques quant à eux sont levés très tôt.

Fonctionnement des méthodes agiles

- Si le client a bien géré les priorités des exigences du produit, il peut accélérer le « **time to market** », s'il estime que le produit en l'état (partiel) peut aller en production. Il économise ainsi son budget et récolte un premier **retour sur investissement**.
- **Il a aussi la possibilité de changer en cours de route** la priorité des fonctionnalités qui n'ont pas encore été développées (prévues pour les itérations futures), afin de retarder une fonctionnalité dont le besoin n'est pas mûr ou ajouter une nouvelle fonctionnalité cruciale en échange du retrait d'une autre (respectant le budget et délais), etc.
- Cette souplesse ainsi offerte est donc un véritable atout pour le client.

Historique des méthodes agiles

- La première approche de gestion de projet de développement itératif date de **1986**.
- La première mise en oeuvre de la **méthode Scrum** (la méthode Agile la plus utilisée, documentée et éprouvée aujourd'hui) date de **1993**.
- La seconde concerne un événement majeur rassemblant, en **2001**, dix sept figures éminentes du développement logiciel pour débattre du thème unificateur de leurs méthodes respectives. De cet événement est né le Manifeste Agile rassemblant à la lueur des expériences de chacun les critères pour définir une nouvelle façon de développer des logiciels.
- Le terme « **Agile** » pour qualifier ce type de méthode est également né à cette occasion.

Historique des méthodes agiles

- Aujourd'hui ces méthodes ont fait leurs preuves. Tout le monde (dans le monde de l'informatique) ou presque a au moins entendu parler d'une **méthode Agile** (Scrum, eXtreme Programming, RAD, Chrystal Clear,...).
- Les outils associés aux méthodes agiles sont disponible sur le marché y compris dans le secteur Open Source.

Le Manifeste Agile

- Le manifeste Agile contient l'essence et la philosophie de l'approche en question. Il illustre à lui seul le changement culturel profond qui est en jeu.
- Il montre comment mieux développer des logiciels par la pratique et en aidant les autres à le faire. Ces expériences ont amené, les auteurs du manifeste, à valoriser :
 - Les individus et leurs interactions plus que les processus et les outils
 - Des logiciels opérationnels plus qu'une documentation exhaustive
 - La collaboration avec les clients plus que la négociation contractuelle
 - L'adaptation au changement plus que le suivi d'un plan
- Les auteurs reconnaissent la valeur des seconds éléments, mais privilégient les premiers.

Principes sous jacents au Manifeste Agile

Les principes du Manifeste Agile:

1. Notre plus haute priorité est de satisfaire le client en livrant rapidement et régulièrement des fonctionnalités à grande valeur ajoutée.
2. Accueillez positivement les changements de besoins, même tard dans le projet. Les processus Agiles exploitent le changement pour donner un avantage compétitif au client.
3. Livrez fréquemment un logiciel opérationnel avec des cycles de quelques semaines à quelques mois et une préférence pour les plus courts.
4. Les utilisateurs ou leurs représentants et les développeurs doivent travailler ensemble quotidiennement tout au long du projet.
5. Réalisez les projets avec des personnes motivées. Fournissez-leur l'environnement et le soutien dont ils ont besoin et faites-leur confiance pour atteindre les objectifs fixés.

Principes sous jacents au Manifeste Agile

6. La méthode la plus simple et la plus efficace pour transmettre de l'information à l'équipe de développement et à l'intérieur de celle-ci est le dialogue en face à face.
7. Un logiciel opérationnel est la principale mesure d'avancement.
8. Les processus Agiles encouragent un rythme de développement soutenable. Ensemble, les commanditaires, les développeurs et les utilisateurs devraient être capables de maintenir indéfiniment un rythme constant.
9. Une attention continue à l'excellence technique et à une bonne conception renforce l'Agilité.

Principes sous jacents au Manifeste Agile

- 10. La simplicité – c'est-à-dire l'art de minimiser la quantité de travail inutile – est essentielle.
- 11. Les meilleures architectures, spécifications et conceptions émergent d'équipes auto-organisées.
- 12. À intervalles réguliers, l'équipe réfléchit aux moyens de devenir plus efficace, puis règle et modifie son comportement en conséquence.

Les principales méthodes agiles à étudier

- **Scrum**
- **XP : eXtreme Programming**
- **Kanban**

Références

1. Introduction aux méthodes agiles et Scrum - Florent Lothon, mis à jour en juin 2013
[agiliste.fr /fiches/introduction-methodes-agiles/](http://agiliste.fr/fiches/introduction-methodes-agiles/)
2. <http://www.agilemanifesto.org/iso/fr/manifesto.html>
3. <http://www.access-dev.com/access-dev/la-gestion-de-projet-methodes-classiques-vs-methodes-agiles/>