

LE CYCLE DE VIE ET MODELE DE DEVELOPPEMENT DU LOGICIEL

Les méthodes agiles

Dr S.Fellah

Introduction

Introduction

- ❖ Le modèle en cascade, universellement connu et utilisé, il se base sur **l'anticipation** des **demandes** des **clients**, la **définition complète** du produit et la **documentation exhaustive**.
- ❖ Les sociétés informatiques sont confrontées à :
 - **fortes pression et concurrence** sur un marché caractérisé par une **diversité** des **offres** technologiques,
 - **instabilité** des **demandes**,
 - **réduction** des **délais** de mise sur le marché.

Limites des modèles classiques

- ❖ **Méthode rigide** : toute la gestion de projet est contenue dans un **plan rédigé en amont** par le chef de projet.
- ❖ **Le manque de communication** et de **collaboration** entre **l'équipe de développement** et **l'équipe métier**.
- ❖ **Le manque de retours d'utilisateurs**.

Introduction

Introduction

- ❖ Depuis les années 1990, les professionnels du développement de logiciels se sont intéressés au concept **d'agilité**.
- ❖ Ils ont tenté d'y apporter une définition.
- ❖ Pour Erickson et ses collègues (2005):

***l'agilité:** incite à répondre très **rapidement**, aux **changements** et aux **contraintes** imposées, par les **délais** toujours plus **courts** de livraison de projets.*

Dans une même optique, d'autres auteurs ont lié le développement «**agile**» aux notions de **flexibilité**, de **rétroaction** et **d'adaptation** au changement **rapide** et **continu**.

Introduction

Introduction

- ❖ De nouvelles **approches**, plus «**flexibles**», **dénommées** «**agiles**», ont été développées.
- ❖ En très peu d'années, **l'agilité** a **connu** un **succès considérable** au niveau de développement de logiciels.
- ❖ Elles **reposent** sur un **cycle** de développement **itératif, incrémental** et **adaptatif**.

Définition des méthodes agiles

Origines des méthodes agile

- ❖ Les premières méthodes agiles voient le jour dans les années 1990 avec :
la publication d'un guide sur la méthode **RAD** (Rapid Application Development) en **1991**.
- ❖ Enfin, viennent d'autres méthodes agiles, telles que **Scrum**, **eXtreme Programming**, **Unified Process** ...

Principes communs aux méthodes agiles

Concepts

- Méthodes **adaptatives**
- **itérations courtes**
- **lien** fort avec le **client**
- **fixer** les **délais** et les **coûts**
- attention à la **communication humaine**
- équipes **auto-organisées**
- Processus **auto-adaptatif**

Principes communs aux méthodes agiles

Méthodes

Méthodes

- Simplicité
- Légèreté
- Orientées participants plutôt que plan
- Nombreuses :
 - ✓ eXtreme Programming
 - ✓ SCRUM
 - ✓ :

Remarque

- Pas de définition unique mais un manifeste

Le Manifeste agile

- ❖ En février 2001, aux États-Unis, rencontre et accord sur un manifeste par 17 spécialistes du développement logiciel.
- ❖ Le **Manifeste** agile est **constitué** de **quatre valeurs** et de **12 principes fondateurs**.

Le Manifeste agile

Les quatre valeurs

- **les individus et leurs interactions** plus que les **processus** et les **outils** ;
- **un logiciel qui fonctionne** 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**.

Le Manifeste agile

12 principes fondateurs

1. La plus haute priorité est de **satisfaire le client** en **livrant rapidement** et **régulièrement** des **fonctionnalités à grande valeur** ajoutée. (**Satisfaire l'utilisateur**)
2. Accueillez positivement les **changements** de **besoins**, **même tard** dans le projet. (**Accepter le changement**)
3. Livrez fréquemment un **logiciel opérationnel** avec des **cycles** de **quelques semaines** à **quelques mois** et une préférence pour les plus courts. (**Livrer fréquemment**)
4. Les **utilisateurs** ou leurs représentants et les **développeurs** doivent **travailler ensemble** quotidiennement tout au long du projet. (**Travailler en groupe**)
5. **Réalisez** les projets avec des **personnes motivées**. Fournissez-leur **l'environnement** et le **soutien dont elles ont besoin** et faites-leur **confiance** pour atteindre les objectifs fixés. (**Stimuler la motivation**)
6. Privilégiez la **colocation** de toutes les **personnes** travaillant ensemble et le **dialogue en face à face** comme méthode de communication. (**Communication directe**)

Le Manifeste agile

12 principes fondateurs

7. Un **logiciel opérationnel** est la **principale mesure de progression** d'un projet.
(**Les fonctionnalités implémentées**)
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**. (**Eviter la précipitation**)
9. Une **attention continue** à l'**excellence** technique et à un bon design.
(**Ne pas négliger la qualité de production**)
10. La simplicité – c'est-à-dire l'art de **minimiser** la quantité de **travail inutile** – est essentielle.
(**Rester concentré sur l'essentiel : mettre en place un produit**)
11. Les meilleures architectures, spécifications et conceptions **émergent d'équipes auto-organisées**.
12. À intervalles réguliers, **l'équipe réfléchit** aux **moyens possibles** pour **devenir plus efficace**. Puis elle **s'adapte** et **modifie** son **mode de fonctionnement** en conséquence.
(**Intégrer la notion d'amélioration continue tout au long du projet**)

Le Manifeste agile

Résumé

- ❖ Le manifeste **ne propose** pas un **mode opératoire**.
- ❖ L'agilité propose une **orientation**, une **philosophie de travail organisée** sur une base de **règles simples**.

LE CYCLE DE VIE DU LOGICIEL

Principales méthodes agiles

- ❖ Les principales méthodes agiles se **nourrissent** des **valeurs** et **principes** du **Manifeste**.
- ❖ Cependant, si **elles ont un tronc commun** de **pratiques**, elles **se différencient** par :
 - le poids de la méthodologie dans la **documentation produite**
 - les **étapes** formelles,
 - le **rythme** du projet ou le **nombre** et la **longueur** des **itérations**.

LE CYCLE DE VIE DU LOGICIEL

Principales méthodes agiles

❖ ASD (Adaptative Software Development)

En 2000, Jim Highsmith (signataire du Manifeste)

❖ Crystal

- Crystal, ou plus exactement la **famille** de **méthodologies** Crystal
- Mise au point par Alistair Cockburn (signataire du Manifeste).
- Les **méthodes agiles Crystal** sont généralement **souples** à mettre en œuvre.
- les méthodes agiles Crystal **ne sont pas centrées sur les processus**.
- Elle sont centrées sur les **personnes**, la **communication** et la **collaboration** entre eux
- Les méthodes agiles Crystal **s'adaptent** à la **taille** de **l'équipe** de développement.
- Une **couleur** est utilisée pour **identifier** le « **poids** » de la méthode agile à **utiliser** en fonction du projet.
- les couleurs : transparent (Clear), jaune, orange, rouge, marron, diamant et saphir.

Crystal Clear pour une équipes inférieures à 8 personnes.

LE CYCLE DE VIE DU LOGICIEL

Principales méthodes agiles

❖ **DSDM (Dynamic Software Development Method)**

DSDM est le fruit du travail d'un consortium de sociétés désirant utiliser RAD de façon structurée et indépendante, en Grande-Bretagne.

❖ **RAD (Rapid Application Development)**

LE CYCLE DE VIE DU LOGICIEL

Principales méthodes agiles

❖ UP (Unified Process)

- Certains **regrettent** sa **lourdeur** et sa **grande similarité** avec un cycle en **cascade**; d'autres, le **considèrent** comme une **méthode agile** très **documentée** et qui, par là même, **permet** une **grande adaptation** au contexte du projet.

- On peut dire qu'il se situe à **l'intermédiaire** des **deux approches** et qu'il est, **grâce** à sa **flexibilité** agile.

❖ XP (eXtreme Programming)

❖ Est une initiative de Kent Beck et Ron Jeffries, **issue** d'une étroite **collaboration** avec Ward Cunningham, expérimentée en **1996** sur un **projet pilote** chez Chrysler.

❖ FDD (Feature Driven Development)

❖ On pourrait également évoquer Xbreed,
❖ EVO (Evolutionary Project Management), d'autres méthodes agiles.

La méthode agile « **scrum** »

La méthode agile « scrum »

Définition

- ❖ Le Scrum ou « mêlée », créée par Ken Schwaber et Jeff Sutherland (signataires du Manifeste).
- ❖ C'est un **cadre** de travail agile.
- ❖ Scrum est un **schéma d'organisation** de développement de **produits complexes**.
- ❖ Scrum **se base** sur des **rôles**, des **réunions**, des **artefacts**, des **règles** de gestion et un **cycle itératif** de développement.
- ❖ On définit le **contenu** d'une **itération** (ou « **sprint scrum** ») en termes de **fonctionnalités**, elles sont **développées**, puis **validées** à l'issue du **sprint**.
- ❖ Un **bilan** du **sprint** écoulé est **réalisé** avant de **continuer** sur le **sprint suivant**.
- ❖ Une Equipe Scrum est composée de **7** personnes.
- ❖ Equipes auto-organisées

La méthode agile « scrum »

Les valeurs de scrum

- ❖ Le scrum est basé sur 3 valeurs :

Transparence

- ❖ Avoir **langage commun** entre l'équipe et le management afin d'obtenir rapidement une **bonne compréhension** du projet.

Inspection

- ❖ Vérifier **l'écart** par **rapport** à **l'objectif initial**

Adaptation

- ❖ **S'adapter** en fonction des **écarts** constatés afin de les **ajuster**.
- ❖ Scrum est favorable à des petits ajustements fréquents.

La méthode agile « scrum »

Les rôles

Le Product Owner

- ❖ Il est l'interface entre le client et les intervenants.
- ❖ C'est une personne qui **joue** le **rôle** du **client** et des **utilisateurs**.
- ❖ Il **définit** les **fonctionnalités** du **carnet du produit** (Product Backlog), et les **priorise**.
- ❖ Il est le seul **habilité** à **prendre** les **décisions** sur l'**orientation** du **projet**.
- ❖ Il est chargé de **maximiser** la **valeur** du **produit** et du travail de l'équipe de développement.
- ❖ Il **valide** les **fonctionnalités développées**.

Development team (L'équipe de développement)

- ❖ Généralement composée de 4 à 6 personnes.
- ❖ **l'ensemble des personnes prenant en charge le projet** : conception, développement, tests, et documentation.

La méthode agile « scrum »

Les rôles

Le Scrum Master

- ❖ Il se charge du **bon déroulement** du **processus**, il est le **leader-serviteur** de l'équipe.
- ❖ Il se **charge** des **réunions** ou Scrum réunion qui ont lieu **tous** les **matins** (l'ordre du jour des réunions scrum ...), **définition** de la **durée** des **Sprints** et **facilite** le travail.
- ❖ Il **facilite** les **interactions** entre les **membres** de l'équipe Scrum.
- ❖ Il veille à la **mise en œuvre** de **l'agilité**.
- ❖ Il **résout** des **problèmes** auxquels seront confrontés aussi bien l'Equipe que le Product Owner.

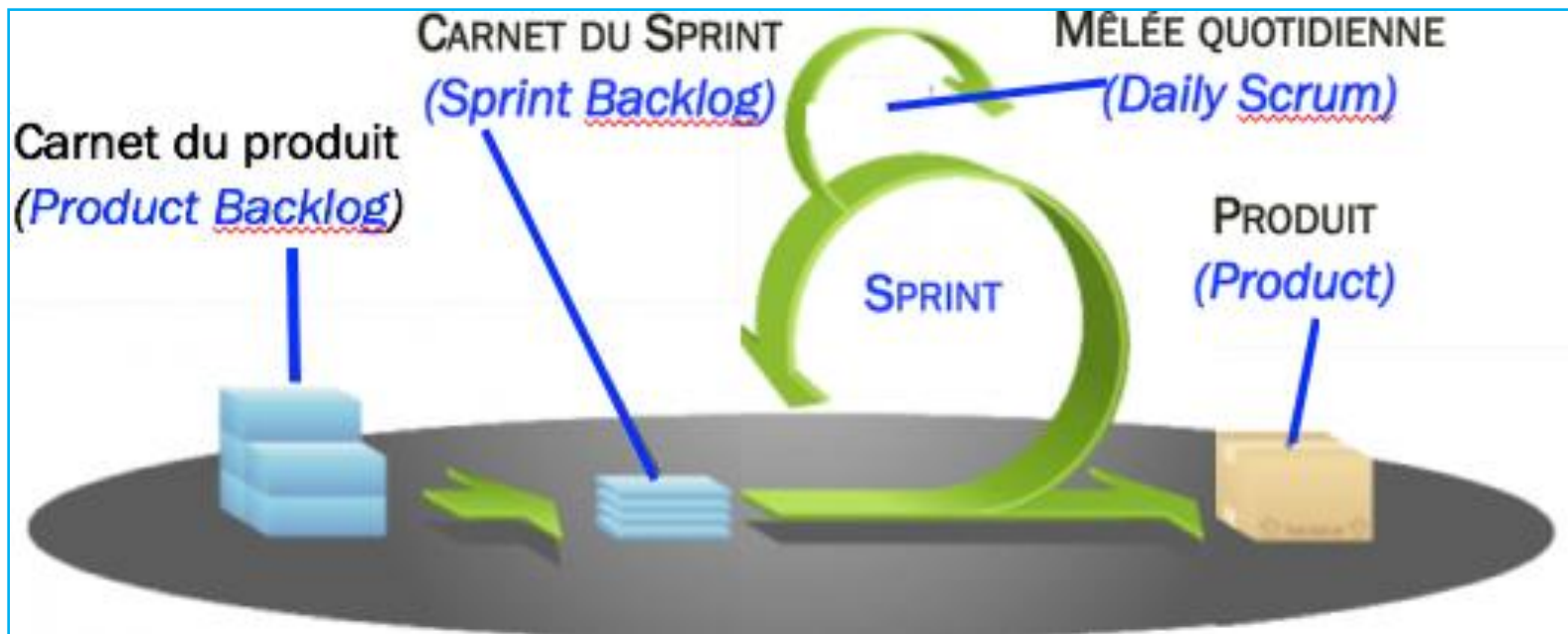
Remarques

- ❖ Le Scrum Master ne dit pas à chacun ce qu'il doit faire et **n'assigne** pas de **tâches**.
- ❖ Il **facilite** le processus, en soutenant l'Equipe dans son organisation et sa gestion.
- ❖ Il n'est pas un chef de projet.
- ❖ Il **n'existe pas de rôle de chef de projet** dans Scrum.

La méthode agile « scrum »

Cycle de développement

- ❖ Le cycle de vie de Scrum est rythmé par des itérations de **quatre semaines** qu'on appelle sprints.



La méthode agile « scrum »

Les événements

Scrum prescrit 4 événements :

- Planification du Sprint (Sprint Planning)
- Mêlée Quotidienne (Daily Scrum)
- Revue de sprint (Sprint Review)
- Rétrospective de Sprint (Sprint Retrospective)

Le sprint

Un sprint a une **durée** de cycle de **deux semaines** à un **mois** au **maximum**, au bout de laquelle l'équipe **délivre** un **incrément** du **produit**.

La méthode agile « scrum »

Les événements

Planification du Sprint (Sprint Planning)

- ❖ Cette réunion définit le **travail à effectuer durant un Sprint**.
- ❖ **Dure au maximum huit heures** pour un **Sprint** d'un **mois**.
- ❖ Un **plan** est **créé** de manière **collaborative** par **tous les membres** de l'équipe.
- ❖ Le Product Owner **discute l'objectif** du sprint fixé et les **éléments** du Backlog Produit **choisis**.
- ❖ Le Scrum Master **veille** à ce que **l'événement ait lieu** et que les **participants en comprennent le but**.

La méthode agile « scrum »

Les événements



La méthode agile « scrum »

Les événements

Revue de sprint (Sprint Review)

- ❖ Le but de la revue est de **montrer ce** qui a **été réalisé** pendant le sprint afin **d'en tirer** les conséquences pour la **suite** du **projet**.
- ❖ Elle est **tenue à la fin** du **Sprint** pour **inspecter l'incrément** réalisé et **adapter** le **Backlog Produit** si nécessaire.
- ❖ **Le Scrum Master** rappelle les **objectifs** du sprint **définis** lors de la **réunion de planification**.
- ❖ L'équipe **présente** le produit réalisé.
- ❖ **Le Product Owner** et les **intervenants** présents posent des questions à l'équipe et **donnent leur avis** sur le **produit réalisé**.

De durée max de 4 heures pour un scrum de 1 mois.

- ❖ Les participants (Le Product Owner, le Scrum Master, l'équipe de développement, membre de l'équipe de management, les utilisateurs invités, personnes intéressées par le projet invitées par Product Owner).

La méthode agile « scrum »

Les événements

Rétrospective de Sprint (Sprint Retrospective)

- ❖ Son but de **tirer les leçons** du **Sprint** afin de les **mettre au profit** des **suivants** .
- ❖ Le but de la réunion est **d'améliorer** le **processus** pour le **prochain sprint**.
- ❖ Son objectif est **d'inspecter l'équipe** elle-même et de **proposer un plan d'action à adopter** dans le **Sprint suivant**.
- ❖ C'est l'occasion de **s'améliorer** (productivité, qualité, efficacité, conditions de travail, etc)

- ❖ **D'une durée de trois heures** pour un Sprint d'un mois.
- ❖ ***Les participants à la rétrospective sont :***
 - L'équipe de développement.
 - Le Product Owner.
 - Le Scrum Master.
 - Éventuellement un membre de l'équipe de management.