**SCRUM**

Table des matières

[Chapitre 1 INTRODUCTION 2](#_Toc95389101)

[Chapitre 2 : METHODOLOGIE AGILE : DEFINITION & AVANTAGES 2](#_Toc95389102)

[Chapitre 3 METHODOLOGIE « SCRUM » 🡺 DEFINITION : 3](#_Toc95389103)

[Chapitre 4 : PRINCIPES &ATREFCATS DE SCRUM : 3](#_Toc95389104)

[Chapitre 5 : LES EQUIPES DE SCRUM : 3](#_Toc95389105)

[5.1 Le Product Owner : 3](#_Toc95389106)

[5.2 Scrum Master : 4](#_Toc95389107)

[5.3 L’Equipe Dev (La Team) : 4](#_Toc95389108)

[5.3 Autres : 4](#_Toc95389109)

[Chapitre 6 : LES EVENEMENTS D’UN SCRUM : 4](#_Toc95389110)

[6.1 Sprint planning (planification de sprint) : 5](#_Toc95389111)

[6.2 Daily Scrum (Daily Meeting) : 8](#_Toc95389113)

[6.3 Sprint Demo : 8](#_Toc95389114)

[6.4 Retrospective Sprint : 8](#_Toc95389115)

# INTRODUCTION

La méthodologie la plus ancienne utilisée pour développer un projet est la méthode cycle en V (vue dans le cours DPI) avec ses différentes phases (environnements) et différentes équipes. Au jour d’aujourd’hui cette méthodologie est délaissée pour ses différents problèmes d’efficacité (qualité du produit final voire échecs), de temps (très lent) et de budgets (ça coute plus cher que ce qui a été prévu).

La **majeure source d’erreur** de cette méthodologie est la distance de temps qu’elle existe entre le début du projet et le produit livré final au client

Donc quand le client final ne voit son projet qu’à la fin il y a de fortes chances que :

* Soit, il ne serve à rien parce qu’il s’est écoulé tellement de choses et le temps était tellement loin qu’on est sorti du contexte du besoin du client.
* Le projet se vire à l’échec car il existe beaucoup de fonctionnalités à développer 🡺 les chances pour aboutir sont minimes.
* Soit, on l’annule complètement parce que le client a changé d’avis entre temps et son besoin a évolué depuis.

**D’autres sources d’erreurs** sont aussi à signaler dans cette méthodologie notamment celle évoquées dans la diapo 9 du slide du cours.

On remarque pour ceci une grande partie des projets informatiques, selon les statistiques vouent à l’échec.

# : METHODOLOGIE AGILE : DEFINITION & AVANTAGES

La nouvelle méthodologie (la solution) « AGILE » est venue en secours à tout ces problèmes pour répondre aux besoins du marché et essayer de minimiser les risques d’échecs (plus efficace, moins couteuse, plus rapide).

###{Cette méthodologie prend toujours en référence l’ancienne méthodologie 🡺 son principe consiste de faire de petits cycles en V à des durées raccourcies}.

On l’appelle « agile » parce qu’elle est **souple** et s’adapte au besoin du projet et on peut faire une réorientation du projet à n’importe quel temps de son évolution 🡺 on présente au client chaque petit compartiment (incrément) du projet grâce à des entrevues organisées régulièrement avec lui, et à chaque réunion il peut valider ou pas le résultat de ce qu’on lui a présenté et surtout ça revient moins cher à l’entreprise (on ne perd pas beaucoup de temps, d’énergie et on dépense moins dans le cas où l’incrément ne correspond pas au besoin client)

Le **principal avantage** de cette méthodologie réside dans son principe fait que les équipes concernées se concentrent essentiellement sur la qualité du produit (sa valeur ajoutée : **vision produit**), et grâce au business modèle de l’application (produit final) on peut optimiser nos ressources et notre temps (argent et temps). **D’autres avantages** aussi à signaler aussi importants comme ceux notés dans la diapo 11 du cours.

**NB :**  il existe différentes méthodologies agiles et nous dans ce cours on aborde LE SCRUM 🡺 la plus demandée sur le marché.

# METHODOLOGIE « SCRUM » 🡺 DEFINITION :

* SCRUM est un terme anglais qui désigne la mêlée (rugby)🡺 ce qui veut dire on dépolit toutes nos forces pour aller dans le même sens.
* Donc le scrum est un état d’esprit qui se base sur la gestion des RH (psychologie, management) qui a comme objectif de fédérer l’ensembles des acteurs de la réalisation d’un projet à penser de la même façon 🡺 il faut donner de la confiance à chaque membre de l’équipe afin d’être plus productif
* Un développeur qui pose directement les questions au client lors d’une entrevue l’aide à cerner au mieux le besoin du client 🡺 supprime le fait qu’on ait des avenants car il n’existe pas d’intermédiaires entre le client et le développeur qui est l’AMOA comme dans la méthode en cycle en V).
* Les principes (avantages en même temps) sur la diapo 14 du cours

# : PRINCIPES &ATREFCATS DE SCRUM :

Le principe du SCRUM consiste à alimenter le **product backlog** (c’est un peu le cahier de charge mais élaboré au fil du temps) par des **US (User Stories),** developper ces US dans **des sprints** (laps de temps maximum 4 semaines) pour avoir **des incréments** livrables en l’état.

* Le **BACKLOG produit** est alimenté par le **PO** (Project Owner) avec les us
* Chaque **US ou ensemble d’US** est développé par l’équipe des développeurs au d’un script.
* **Definition of done** : la définition commune à toute l’équipe à partir de quand le but est atteint.
* **Backlog sprint**: c’est un ensemble des us qu’on implémante dans un sprint

# : LES EQUIPES DE SCRUM :

On distingue 3 types +(1) de d’intervenants axiaux(élémentaires) dans le la méthodologie Scrum ; à savoir :

* LES PO (Product Owner)
* Le Scrum Master
* L’équipe DEV (La Team)
* autres

## 5.1 Le Product Owner :

* C’est le propriétaire du produit 🡺il comprend le besoin du client 🡺 c’est lui qui garantit la valeur du produit 🡺 il faut qu’il soit toujours en contact avec le client 🡺 c’est lui qui se charge de rédiger les **spécifications fonctionnelles (US = parcours = cas d’usage= récit** /plus est petite plus a de chances pour mieux réussir**)** sous forme de scénarios (comme un film)

**Exemple d’US :**

Sur une page de logins d’un site internet changer le mot de passe est un us

Faire un virement est un us ; faire un virement instantané est une us, programmer un virement est une us.

* Donc le rôle de du PO sert aussi à alimenter le backlog avec les us en respectant la priorité.
* Dans son rôle aussi de réaliser et de valider le résultat d’un sprint en réalisant les **tests d’acceptance (**test qui dit à quel moment on accepte un us ; même s’il y a des réserves à mettre sur quelques aspects périphériques (aspects ergonomiques d’une application par exemple en réserve).

## 5.2 Scrum Master :

* C’est celui qui veille à pratiquer la méthodologie scrum dans un projet. C’est un coach, psychologue, (prédicateur/anticipateur de la capacité de l’équipe du développement). Il doit s’entendre avec tout le monde avec le sourire pour avoir des conditions de travail optimales dans la réalisation du projet.
* Il est au service du PO et au service de l’équipe dev. (la Team)

**NB :** Un chef de projet dans la méthode en V est un bon candidat pour devenir un un Scrum Master

UN AMOA est un bon pour devenir un bon PO.

## 5.3 L’Equipe Dev (La Team) :

* Se compose de profils dont on a besoin pour implémenter les us pour les passer en prod. ; elle peut donc renfermer des testeurs (agile testing), des développeurs, etc selon le besoin.

**NB :** si on n’a pas suffisamment d’us à implémenter dans un sprint alors on use la dette technique.

## 5.3 Autres :

* Comme les PO sont très chargés et leur travail est tellement crucial dans le long du projet on leur ajour alors d’autres assistants qui sont les **BA (Business Analysts**) qui sont sous leur responsabilités et à qui ils confient des tâches.

**NB :** Le business analyst peut faire toutes les tâches du PO qu’il le gère.

* Dans une DSI comme il est tellement difficile de faire évincer les anciens chefs de projets et les AMOA qui exerçaient depuis un moment alors on peut tomber sur Un AMOA ou un chef de projet par exception
* On peut aussi travailler dans un projet où l’agilité est appliquée seulement au niveau de l’équipe AMOE tant dis que le reste des équipes sont sous l’ancien système.

# : LES EVENEMENTS D’UN SCRUM :

Le scrum est le sprint et dedans On peut noter 4 types d’évènements :

**1-Planification du sprint** : réunion avant de débuter un sprint on décide des us à engager

**2-daily scrum** : tous les matins (rituel) : ce que j’ai fait, ce que je fais aujourd’hui, et les problèmes rencontrés

**3-sprint demo** : présence du client, pour leur présenter ce qu’on a fait.

**4-rétrospective** de sprint : s’autocritiquer en équipe pour mieux améliorer nos produits. C’est souvent en entreprise on peut intervenir sur stade en tant que développeur pour être force de proposition.

**NB :** le 1er sprint est le sprint 0 c’est un sprint de cadrage, on ne se fixe pas de temps particulier (1 semaine, 2 semaines, 3 semaines, 7 semaines…) c’est à ce niveau qu’on se décide des lignes directrices du projet on recrute le personnel nécessaire pour la Team. Généralement il n’y a que le PO et le scrum master qui sont concernés. (On ne décide pas de la durée des sprint).

## 6.1 Sprint planning (planification de sprint) :

Dure en moyenne 1,5 🡺 2h / 1 semaine de sprint ; L’équipe PO, le scrum master et la team y assistent. C’est à ce stade qu’on s’occupe de lire (l’équipe Po avec explications et réponses aux questions si besoin dans le but de s’assurer que la team a bien compris ce qu’elle devra faire le long du sprint et pourra donc répondre aux attentes du client) et d’écrire sur confluence (ATLANSIAN) les us. A chaque fois qu’on clique sur une us elle nous renvoie vers une autre page.

Les us viennent essentiellement du product backlog mais elles peuvent aussi venir du reste des us non-réussites du précédent sprint et dans ce cas-là sont surement prioritaires.

On se charge aussi de mettre à jour jira (le tableau de bord) (déplacer les étiquettes vers doing, done… etc).

## 

Planning sprint

Une fois que la team ait compris ce qu’elle devra faire lors du prochain sprint (les us à faire), elle se réunit (avec présence ou absence de l’équipe PO généralement ne sont pas présents car c’est très technique mais ils peuvent être présents selon les projets les sociétés) pour décomposer chaque US en plusieurs taches.

Une fois chaque us concernée par le sprint est décomposée en tâches on passera au poker planning (jeu de carte) pour estimer la capacité de l’équipe.

Le poker planning consiste à donner un degré de complexité à chaque tâche d’une us par chaque membre de l’équipe y compris le scrum master.

Le but de sprint planning est de connaitre la vélocité de l’équipe dans le projet par le scrum master.

**Exemple :** la page loging d’une applicatin, on prend l’us cliquer sur ok est une action (évènement)

Code python ou java

Base de données ou AD

OU SSO

Login user

mdp

ok

webservice

**backend**

**IHM**

US 125 : titre loging page

* IHM (T145)
* Habillage css (T146)
* Gestion des codes retour (si ok accès connexion sinon pas de connexion T147))

ENSEMBLE DES TACHES DE L’US 125

* API test T148
* Interface annuaire T149
* Gestion des logging T150

Chaque tâche génère un ticket (postite) qui ressemble à la figure suivante :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| US 125 | | T145 | | |
| Description de la tache | | | | |
| Date | Nbre de pts de complexité | | SSE (la personne de l’équipe qui se charge de la faire) | |
| TO DO (à faire) | | | | DOING (en cours de réalisation) | | DONE (réalisée) |
| US 125  T145, T146, T147, | | | |  | |  |

SENS D’AVANCEMENT DU SPRINT

Le graphique suivant (Burn down charte) est réalisé par le scrum master pour suivre la vélocité de son équipe au niveau de chaque sprint.

**NB :**  Trello et Jira sont des outils très utilisés en entreprise dans l’agilité.

Une us transverse est une us qui est en lien avec une autre. Comme les us de la charte graphique par exemple ; une us de nomenclature des autres us aussi.

.MD = Marc down est le langage de git.

Actif Directory est un serveur d’identité windows.

## 6.2 Daily Scrum (Daily Meeting) :

C’est un évènement Quotidien le long de chaque sprint (début jusqu’à la fin. Le but est pour chacun des membres de la team dise (brièvement) ce qu’il a fait la veille et qu’il compte faire au cours de la journée en cours.

## 6.3 Sprint Demo :

Une demi-journée avant la demo toute la team s’arrête de travailler sur le développement et se concentre sur la préparation de de la demo pour l’équipe PO et pour le client (s’il sera).

L’équipe donc se met à faire des tests de préprod, tests croisés, préparer les jeux de données, chercher tous les cas possibles qui peuvent nuire à leur incrément et les corriger rapidement et efficacement avant le jour J.

Quand le jour J arrive l’équipe PO débarque dans une salle en réunion (Scrum master qui la convoque) avec la Team et on leur fait la démo de ce qu’on a fait le long du sprint (Incrément) en une durée moyenne de 1,5 à 2h / 1 semaine de sprint.

C’est à ce niveau que l’équipe PO se charge de tester à leur façon et comme ils veulent l’incrément livrable. C’est aussi l’occasion pour eux de critiquer de valider ou pas le livrable avec des réserves émises sur chaque US. Dans le cas une us ou une tâche de l’us n’est pas acceptée elle sera directement redirigée vers le prochain backlog sprint et elle sera à ce moment-là prioritaire.

## 6.4 Retrospective Sprint :

C’est comme un jeu, chacun essayera d’une façon anonyme de donner une note au sprint qui vient de se terminer afin de s’améliorer lors des prochains sprints.

Le but est de s’autocritiquer pour mieux améliorer les futures sprint et c’est aussi le but de la méthodologie scrum.

**NB :**

* **RC =** Release candidate**:** version qu’on peut publier.
* **Un job :** c’est un programme qui n’est pas transactionnel (s’exerce en tâche de fond) (absence d’IHM, pas de relation avec l’humain ; il existe différents types d jobs :
* Job de migration
* Job jetable
* Job one chotte
* D’archivage (déplacer des données vers une autre BD)
* Job de purge (supprimer des données dont on n’a pas besoin)
* Job de calculs ( de prime, de salaire, etc)
* Système opérationnel : c’est toutes les applications d’une entreprise qui participent à son business modèle. (génère quotidiennement de la data qui sera utilisée par le système décisionnel)
* Système décisionnel : c’est un système qui aide le personnel de l’entreprise à prendre des décisions et prédire des phénomènes futurs. (il se base essentiellement sur la data générée par le système opérationnel).
* Data Warehouse : c’est l’endroit où se concentre l’ensemble de la data.
* Data mark : c’est une partie de datawirhouse.