Projet dirigé

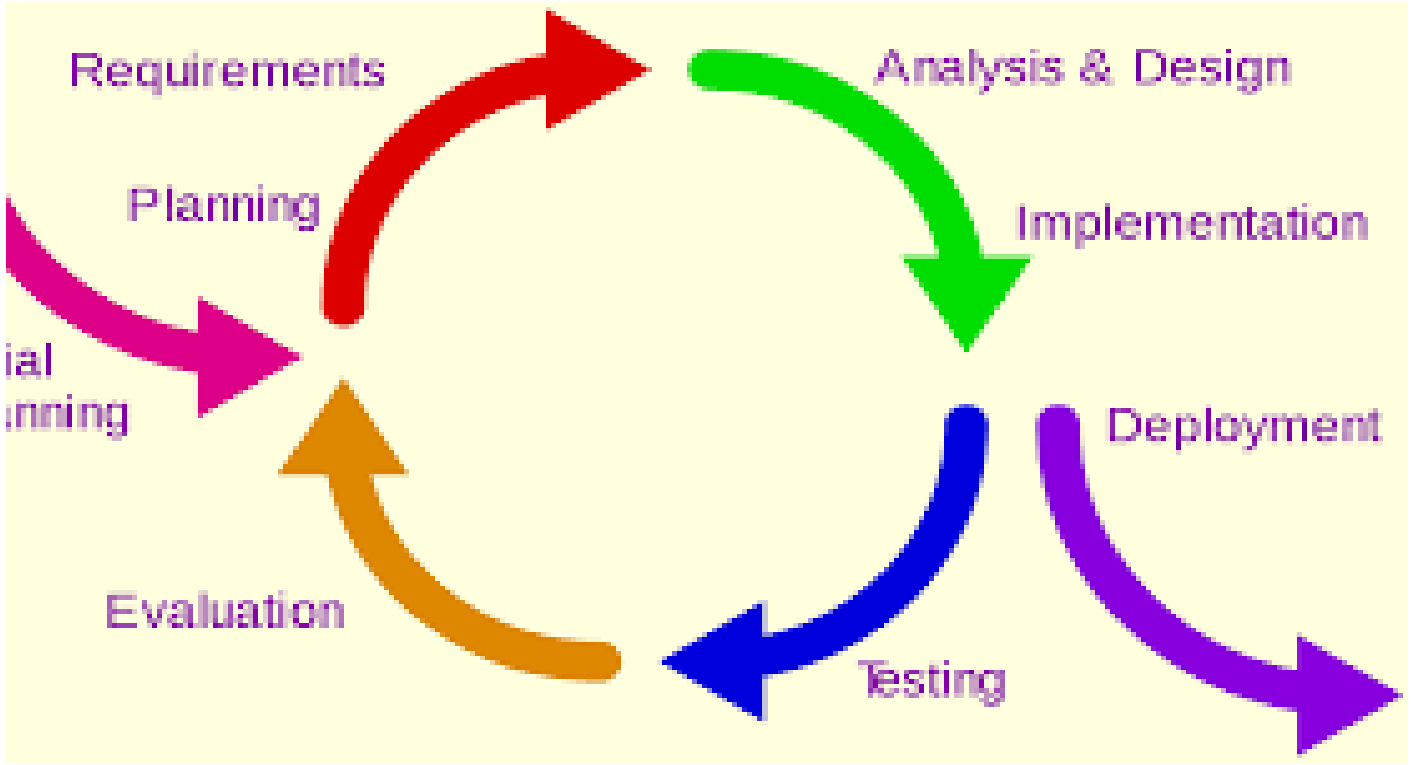
SCRUM, introduction

SCRUM,

Plan de la séance

- Retour sur les user-stories
- L’approche SCRUM
- Cycle de développement selon SCRUM
- Le backlog du produit.
- L’équipe SCRUM
- Prioriser une story
- Estimer une story
- Plan de Release
- Le sprint zéro.

Rappels, l’approche agile



Rappels

Avantages:

- Itératif et incrémental pour vrai pas uniquement dans le principe.
- Livraison tôt et souvent
- Implication du client: les besoins évoluent, feedback tôt.
- Les itérations (sprint) sont courtes.
- La documentation n’est pas lourde.
- Le rythme de travail est constant

Rappels

Manifeste agile: 4 valeurs et 12 principes

Valeurs agiles

1. Les personnes et leurs interactions sont plus importantes que le processus et les outils
2. Un logiciel qui fonctionne prime sur la documentation
3. La collaboration avec les clients est préférable à la négociation contractuelle
4. La réponse au changement passe avant le suivi d'un plan.

Rappels

Les principes : Le Manifeste annonce douze (12) principes, qui ne définissent pas une méthode agile.

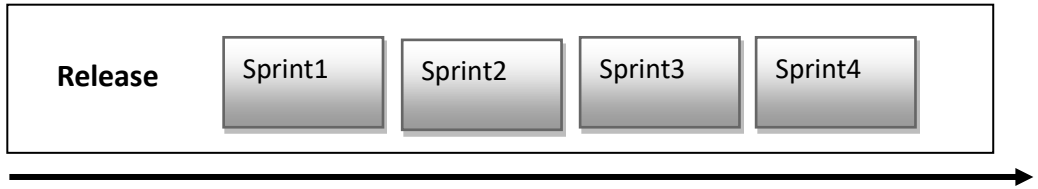
1. Satisfaire le client en livrant tôt et régulièrement des logiciels utiles qui offre une véritable valeur ajoutée.
2. Accepter les changements même tard dans le développement.
3. Livrer fréquemment une application qui fonctionne.
4. Collaborer quotidiennement entre clients et développeurs.
5. Bâtir le projet autour de personnes motivées en leur fournissant environnement et support, et en leur faisant confiance.
6. Communiquer par des conversations face à face.
7. Mesurer la progression avec le logiciel qui fonctionne.
8. Garder un rythme de travail durable.
9. Rechercher l'excellence technique et la qualité de conception
10. Laisser l'équipe s'auto-organiser.
11. Rechercher la simplicité.
12. À intervalle régulier, réfléchir aux moyens de devenir plus efficace.

SCRUM, l'approche

- SCRUM signifie mêlée en rugby.
- Scrum utilise les valeurs et l'esprit du rugby et les adapte aux projets de développement. Comme le *pack* lors d'un ballon porté au rugby (faire progresser le ballon vers l'avant), l'équipe chargée de développement travaille de façon collective et soudée vers un objectif précis
- SCRUM est l'approche agile la plus populaire
- Définitions:
  - Un **sprint** est le terme utilisé dans SCRUM pour désigner une itération. Dans le langage SCRUM un sprint est un bloc de temps fixée aboutissant à un incrément de produit potentiellement livrable.
  - Une **version** est produite par une série d'itérations d'un mois, parfois même de 15 jours, appelés **sprint**. Le contenu d'un *sprint* est défini par l'équipe avec le Product Owner, en tenant compte des priorités et de la capacité de l'équipe.
  - **Une release** : (selon le dictionnaire du jargon informatique) : Version d'un système, par exemple un logiciel, effectivement publiée, donc lâchée dans la nature.
  - **Backlog** du produit: liste unique des user-stories **classées par priorité.**

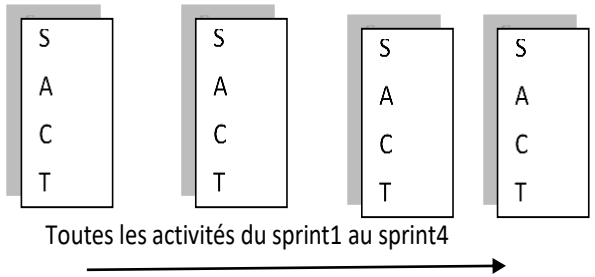
Un sprint trop court risque de stresser l'équipe. Un sprint trop long risque de démotiver l'équipe

SCRUM, l’approche



- Spécification fonctionnelle (Requierements ou étude des besoins fonctionnels)
- Architecture (conception)
- Codage (et test unitaire)
- Test (test d’intégration, test d’acceptation).

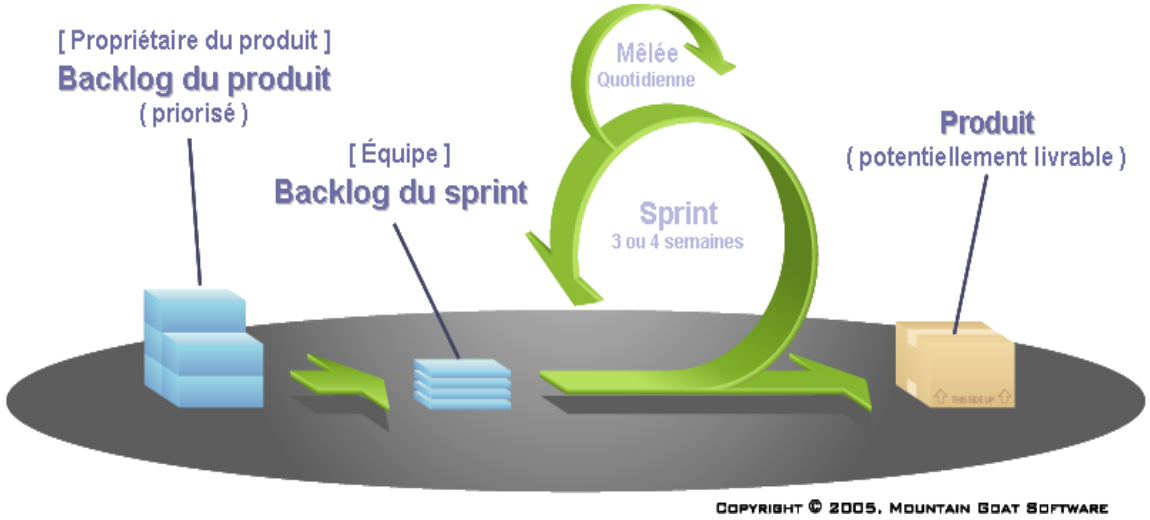
Cycle SCRUM : les sprints et leurs phases se déroulent en parallèle.



SCRUM, l’approche

- Il n’y a pas de chevauchement. Les *sprints* s’enchaînent sans délais. Un *sprint* n’est commencé que si le précédent est terminé. Le nouveau *sprint* démarre immédiatement après le précédent
- À la fin de chaque *sprint,* l’équipe obtient un **produit partiel**, (qui s’enrichit d’un nouveau incrément à chaque *sprint*) **qui fonctionne**. Il est potentiellement livrable. L’ évaluation et le *feedback* récolté permettent d’ajuster le *backlog* pour le *sprint* suivant.
- La date de fin d’un *sprint* est fixée au début de celui-ci. Elle ne change pas même si l’équipe ne réalise pas tout ce qu’elle pensait faire
- Arrêter un sprint plutôt que de l’étendre.

SCRUM, l'approche



Source de l'image : https://fr.wikipedia.org/wiki/Scrum_(d%C3%A9veloppement)

SCRUM, l’approche

- Pour une équipe, une release dure environs 3 mois avec des **sprints de deux à trois semaines**. Ce qui permet d’avoir de quatre à 6 sprints dans une release.
- Il n’y a pas de chevauchement entre les sprints. Ils s’enchaînent sans délais.
- La fin d’un sprint peu être un produit potentiellement livrable.
- Le résultat d’une *release* est le produit livrable fourni à ses utilisateurs. La façon dont il est fourni dépend de son déploiement.
- Souvent, le jalon majeur que représente la *release* correspond à une annonce marketing.

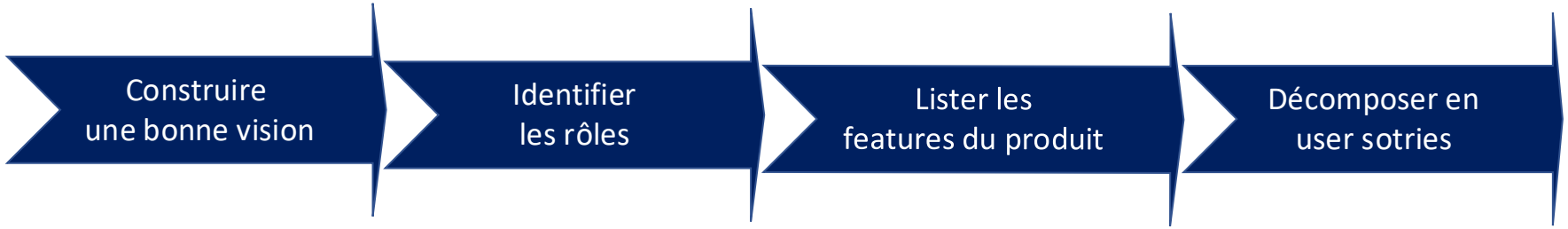
SCRUM, l'approche

SCRUM…. Développement léger

- **Création du *backlog*** de produit (un To-Do list) de toutes les fonctionnalités d'un projet. Les éléments du backlog doivent être estimés (temps de réalisation) et priorisés.
- **Création d'un sprint backlog** : fonctionnalités à compléter durant la durée du sprint (15 jours ou un mois)
- **Effectuer des rencontres quotidiennes durant le sprint** : des mêlées quotidiennes (Daily *scrum*). Pendant un *sprint,* des points de contrôle sur le déroulement des tâches sont effectués lors des mêlées quotidiennes (*scrums*). Cela permet au « Scrum Master », l'animateur chargé de faire appliquer Scrum de déterminer l'avancement par rapport aux engagements et d'appliquer, avec l'équipe, des ajustements pour assurer le succès du *sprint*.
- **Finalisation du sprint** avec démonstration et évaluation. L' évaluation et le *feedback* récolté permettent d'ajuster le *backlog* pour le *sprint* suivant.

SCRUM, le backlog

- Le cœur de SCRUM est le *Product Backlog ou* ***backlog du produit*** qui représente la liste des requis **priorisés. C'est un tableau de user-stories.**
- La production du backlog du produit n'est pas une tâche facile à faire. La question est comment passer d'une vision à backlog de produit ? (voir image plus bas)
    - Construire une bonne vision: La meilleure façon d'identifier les éléments d'une vision est de procéder par des séances de travail en groupe : **un *brainstorming* collectif**
    - Identifier les rôles par l'identification des acteurs
    - Lister les feature: Une *feature* est un service fourni par le système, observable de l'extérieur qui répond à un besoin et dont la description se situe à un niveau tel que toutes les parties prenantes comprennent facilement ce dont il s'agit
    - Décomposer en user-stories:
        - En travaillant en équipe.
        - Éviter des stories trop grosses
        - Éviter d'avoir trop de stories au début.



SCRUM, le backlog

- L'usage le plus courant est de définir un élément du backlog comme étant une *User-Story*
- **Dans un backlog de produit, les stories sont rangées (Classées) selon l'ordre envisagé pour leur réalisation (priorité). Cette notion priorité prend une grande importance dans le développement itératif.**
- SCRUM débute avec un produit **backlog priorisé.**
- Chaque entrée du backlog représente une user-story décrite dans le langage et la terminologie du client. Chaque entrée sert à finaliser ce que le client désire obtenir.
- Le Backlog doit être un document partagé, détenu par le PO.
- Le backlog du produit est vivant.
- Garder le **produi**t **backlog** niveau métier. Il doit focaliser sur les buts métier et non les technologies.

SCRUM, le backlog

Exemple de Backlog

id	Énoncé	Test d'acceptation	Priorité	Estimation
1	En tant que client de la banque , je veux me connecter au site pour choisir une opération bancaire	Connexion réussie Menu d'option affiché	M	2
2	En tant que client, je souhaite consulter mes transactions du mois afin de connaître ma situation financière.	Selon la période choisie: (semaine, mois, trimestre) la liste des transactions est affichée	M	5
3	En tant que client je souhaite réaliser une opération bancaire	N'est pas une bonne User-Story		

- Une colonne « détails »pourrait être ajoutée

SCRUM, le backlog

- **Ce qu'il faut faire :**
  - Cultiver le backlog : le backlog évolue dans le temps, il faudra le mettre à jour.
  - Partager le backlog avec toute l'équipe
  - Surveiller la taille du backlog : ne pas avoir plus de 150 éléments à faire dans le backlog

- **À Éviter :**
  - D'avoir plusieurs backlog pour le même de produit
  - De ne pas avoir de backlog
  - De confondre le backlog de produit avec le backlog de sprint

SCRUM, rôles et responsabilités

Le Product Owner , (PO) :

Le PO est l'expert du domaine (niveau métier). En tant que représentant des clients et utilisateurs, il est responsable de définir les caractéristiques du produit développé par l'équipe, en termes de :

- Fonctionnalités offertes. Plus précisément, il identifie chaque exigence que doit satisfaire le produit et la collecte comme élément du backlog de produit. Il est souhaitable d'inclure les tests d'acceptation.
- Priorité. C'est lui qui définit l'ordre dans lequel ces éléments seront développés en fonction de la valeur qu'ils apportent aux clients et utilisateurs. Cela permet d'alimenter l'équipe avec un backlog de produit prêt pour la planification des sprints
- But. C'est lui qui définit l'objectif d'une release et qui prend les décisions concernant le planning de la release.
- Son implication dans le projet est capitale pour la réussite de celui-ci.

SCRUM, rôles et responsabilités

Le SCRUM Master :C’est le coach de l’équipe (ancien chef de projet). Il a pour rôle:

- Dans le cadre du développement d'un produit, d'aider l'équipe à travailler de façon autonome et à s'améliorer constamment. Il est le garant de l'application du processus, Scrum en l'occurrence.
- S’assurer que l’équipe bénéficie des meilleures conditions pour accomplir les tâches
- Éliminer les obstacles : prendre en compte les problèmes qui surviennent à tout moment sur un projet pour les éliminer au plus vite, en évitant qu'ils ralentissent l'équipe. Il protège l’équipe des interférences extérieures.
- Faire en sorte que l’équipe reste concentrée sur le véritable objectif du projet, qui est de réaliser les éléments du Backlog en collaboration étroite avec le Product Owner , et soit productive. Il s'assure que chacun participe pleinement aux travaux de l'équipe.
- Organise et anime les réunions qui constituent le cérémonial.

SCRUM, rôles et responsabilités

Exemple de rencontres SCRUM: rencontre pour chaque itération (Sprint) dans le processus SCRUM.

É: Équipe **SM:** Scrum Master **PO:** Product Owner

H: Haute direction et le client

Rencontres	Durée (environ)	Entre	Gérée par
Création Sprint backlog	1 journée	PO, É	SM
Scrum quotidien	15 mn	É, SM	SM
Revue sprint (Démo)	4 heures	PO,É,S', H	SM, É
Rétrospective de sprint	3 heures	É,SM	SM

SCRUM, prioriser les éléments du backlog

- **Prioriser les éléments du backlog de produit → Questions à se poser:**
  - Quel est le bénéfice financier ou la valeur ajoutée à développer cette fonctionnalité ?
  - Quel est le coût de développement ? De maintenance …
  - L'implémentation de la fonctionnalité nous permet-elle d'apprendre ?de développer de nouvelles compétences ?
  - La fonctionnalité va-t-elle améliorer la productivité de mon personnel ?
  - Quel est le préjudice si cette fonctionnalité n'est pas implémentée ?

- **Tenir compte de:**
  - Le bénéfice financier attendu ou valeur ajoutée, c'est l'élément le plus important à considérer.
  - Le coût de développement
  - L'opportunité d'apprentissage pour l'équipe
  - Le risque de développement, le développement de cette fonctionnalité nous expose t-elle à d'avantage de risques.

SCRUM, MoScoW

Prioriser les éléments du backlog de produit: MoSCoW

La technique utilisée pour prioriser les besoins dans un contexte itératif est celle de MoSCoW. L'avantage de la méthode *MoSCoW* réside dans la signification de l'acronyme, qui est plus compréhensible que d'autres techniques de priorisation comme élevé/moyen/faible

- **M** pour **Must Have** : ***DOIT*** être fait. L'exigence est essentielle. Si elle n'est pas faîte le projet échoue. On peut dire également priorité haute.
- **S** pour Should **Have** : Il s'agit d'une exigence essentielle, qu'il faut faire dans la mesure du possible (***DEVRAIT***). Mais si elle n'est pas faîte, on peut la contourner et la livrer plus tard.
- **C** pour **Could Have**: Il s'agit d'une exigence souhaitable. Elle ***POURRAIT être*** faîte dans la mesure où elle n'a pas d'impact sur les autres tâches
- **W** pour **Won't Have** Il s'agit d'une exigence «Luxe». ***NE SERA PAS*** faîte cette fois mais plus tard, mais intéressante et à garder pour la prochaine version

SCRUM, priority Pocker

Prioriser les éléments du backlog de produit: priority Poker

- Chaque participant reçoit un lot de neuf cartes numérotées de 1 à 9
- Chaque story est étudiée successivement
- Le premier vote porte sur l'intérêt d'avoir le feature. Chaque participant vote avec une carte. On fait le total des points.
- Le deuxième vote, porte sur la pénalité de ne pas avoir le feature dans le produit. On vote également de 1 à 9.
- On définit l'importance des deux votes. On peut donner 4 au premier et 1 au deuxième.
- En faisant la somme des deux votes pondérés on obtient l'utilité de l'élément. Les stories les plus utiles sont les plus prioritaires.

SCRUM, estimation d'une story

Comment estimer une story ?

- **Demander à un expert:**

c'est bien mais le problème c'est qu'il va se baser sur son expérience, sa capacité. Dans les approches agiles, le développement se fait en équipe. Parfois il est difficile d'avoir un expert dans l'équipe.

Demander à Usain Bolt le temps à mettre pour courir le 100 m ?

- **Analogie:**

Par analogie avec ce que vous connaissez : c'est un peu ce que nous faisons d'habitudes. On dira qu'une story est un peu plus longue que la story précédente. Ou qu'une story A prendra deux fois plus de temps qu'une story B..

- **Découper et détailler.**

C'est plus simple de dire qu'une fonctionnalité ou une petite story sera implémentée dans 3 jours que de dire que la story sera implémentée dans 102 jours ??

SCRUM, estimation d'une story

Mike Cohen(un des contributeur à SCRUM) à propos de l'estimation des user-stories :

- Prenez une équipe composée d'un chirurgien du cerveau et d'un enfant. Le backlog de produit contient les stories suivantes :
    - En tant qu'enfant, je dois coller 1000 timbres sur 1000 enveloppes
    - En tant que chirurgien, je dois faire une opération simple au cerveau.
- Quelle est la story qui nécessite le plus de points ?

Il est probable que les deux stories se terminent en même temps. Que les deux stories aient besoin d'un même nombre de points pour se terminer.

Mais, si la question est : Quelle est la story la plus complexe? La réponse et claire : celle du chirurgien

SCRUM, le planning Poker

Comment estimer une story ?le planning Poker

Pour commencer la séance de planification, il suffit de choisir une story connue de TOUS , (baseline) pour laquelle l'équipe décide en commun de lui fixer une valeur. Et, il est préférable e choisir une story de taille moyenne (3 ou 5) pour laisser une marge vers le bas et vers le haut

1. Le PO présente la story.
2. Les membre de l'équipe posent des questions pour clarifier la story.
3. Tous les participants présentent en même temps la carte choisie pour l'estimation
4. L'équipe discute des différences éventuelles entre les estimations.
5. On recommence jusqu'à une convergence des estimations.
6. On passe à la prochaine story.

Attention: complexité ne veut pas dire plus de points. (plus long)

Les chiffres sur les cartes sont: : 0, 0.5, 1, 2, 3, 5, 8, 13, 20, 40, 100, « ∞ » et « ? ».

SCRUM, le planning Poker

Comment estimer une story ?

Les valeurs des cartes ne représentent pas une charge en termes de jours-hommes, et donc pas directement une durée de réalisation. Ces valeurs sont des points prenant en compte la durée de réalisation d'une tâche, mais surtout sa complexité seule, et aussi par rapport aux autres fonctionnalités demandées. Il s'agit uniquement d'un nombre de points.

- Une valeur de 0 représente une fonctionnalité déjà mise en place, ou ne demandant pas d'effort particulier.
- Les valeurs 0.5 et 1 peuvent être utilisées pour des tâches particulièrement simples,
- 3 et 5 pour des tâches un peu plus complexes.
- Au-delà, la réalisation demande des travaux plus conséquents ou plus complexes.
- La valeur 100 représente une grande complexité et un grand nombre d'heures de travail. Il faudra découper
- La carte portant le symbole « ∞ » (infini) représente toute tâche valant plus de 100. Il s'agit donc d'un travail particulièrement long ou complexe nécessitant une attention particulière. Généralement, une fonctionnalité recevant cette note a peu de chance de pouvoir être embarquée dans le sprint à venir.

SCRUM, le planning Poker

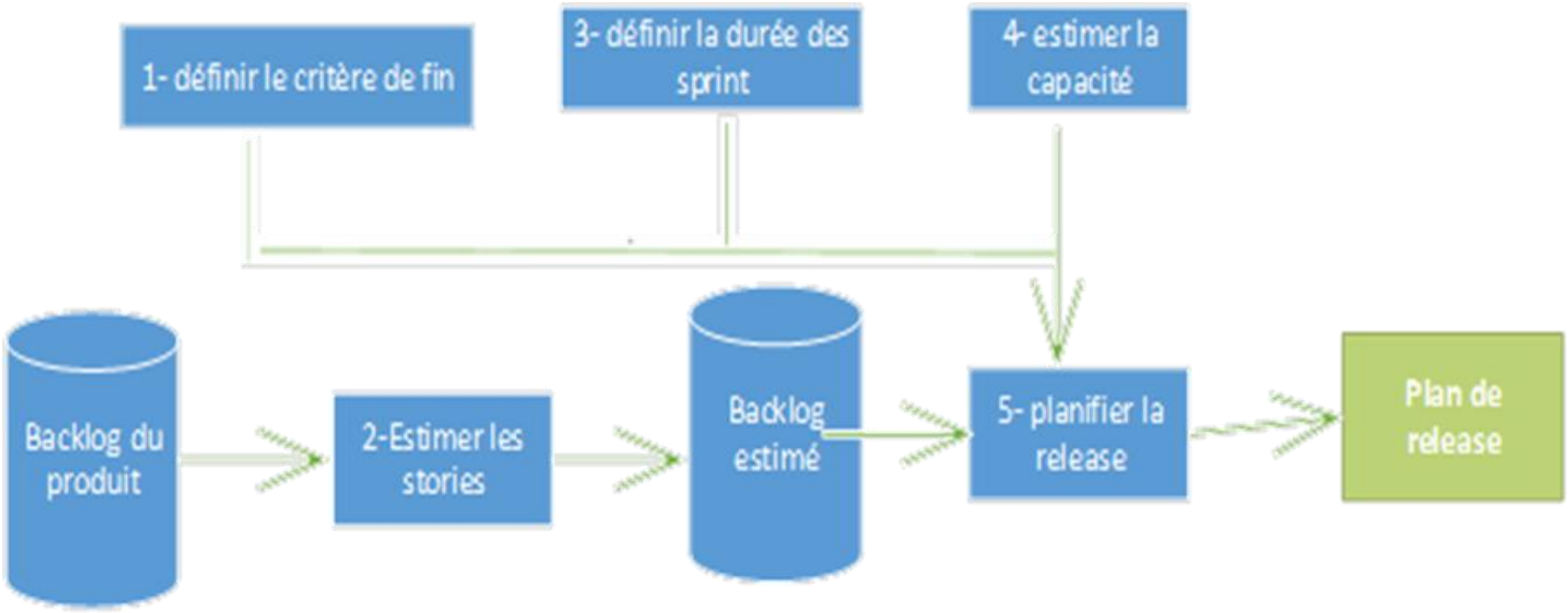
Exemple de cartes?

SCRUM, planifier son projet

Planifier son projet avec SCRUM

- Une release est une séquence de sprints,
  - mais quand finit-elle?
  - Comment la planifier pour livrer un produit fini et à temps ?
  - Combien de sprints la release contiendra-t-elle ?
  - Quelle est la durée d'un sprint ?

SCRUM, planifier son projet



SCRUM, définir la date fin.

Étape 1: définir la fin de la release :

- Quand le backlog est vide. Mais le backlog est vivant
- Fixer les éléments à livrer à l'intérieur du backlog pour la Release courante. Les autres éléments seront livrés dans une nouvelle version: C'est ce qu'on appelle une Release à paramètres fixés. (le backlog étant priorisé→on livre les éléments prioritaires en premier
- Fixer la date fin à l'avance: cette façon de faire permet de fixer les éléments à livrer compte tenu de la date de livraison.

SCRUM, définir la date fin.

Avantages d'une release à date fixe

- Elle donne un objectif précis pas lointain ---> motivation de l'équipe
- Elle impose au Product Owner d'avoir une réflexion poussée sur les priorités des éléments du backlog
- Des éléments du backlog ayant peu d'intérêt ne seront pas développés.
- On passe moins de temps à planifier puisque la date de livraison est connue.

Une release à date fixée et à durée uniforme (exemple une durée de 3 mois) est la formule la plus facile à mettre en œuvre.

SCRUM, estimation

Étape 2:Estimer les stories

- L’estimation se fait en équipe: exemple du planning Poker

SCRUM, la durée de sprint

Étape 3:Définir la durée de sprint

- Lorsqu'on utilise une approche agile pour le développement logiciel, la question fondamentale est de déterminer la durée d'un sprint. Chaque projet est différent. Il n'y a donc pas de réponse universelle.
- Pour SCRUM, la pratique est de faire des sprints de maximum UN MOIS. Dans la plupart des cas, des sprints de 2 à 3 semaines sont recommandés.
- Un sprint trop long risque de démotiver l'équipe
- Un sprint trop court… risque de mettre plus de stress su l'équipe.

SCRUM, la durée de sprint

Pour définir la durée d'un sprint, on tient compte:

- L'implication des clients et des utilisateurs : Il faut tenir compte de leur disponibilité à utiliser les versions partielles produites à la fin de chaque sprint
- Le coût supplémentaire engendré par la préparation du sprint : Un sprint ajoute du travail supplémentaire pour préparer le produit partiel. Faire les tests de non régression, préparer la démonstration pour la revue de sprint
- La taille de l'équipe : plus il y a du monde dans l'équipe, plus il faudra du temps pour se synchroniser.
- La date de fin de la release : idéalement une release comporte au moins quatre sprints pour profiter des bénéfices des avantages de l'itératif
- La stabilité de l'architecture: il est plus facile d'obtenir un produit qui fonctionne si l'architecture est stable. (conception globale)

SCRUM, la vélocité

Étape 4:Estimer la capacité de l'équipe

Définitions : La vélocité.

- La Vélocité: la vélocité de l'équipe **mesure** la partie du backlog **réalisée** par l'équipe à l'intérieur d'un sprint. À la fin d'un sprint, on mesure ce que l'équipe a été capable de réaliser.
  - La vélocité se calcule à la fin d'un sprint après la démonstration et lors de la revue de sprint
  - La vélocité est une mesure de l'équipe et non d'une personne individuelle.
- La capacité: La capacité est une prévision de ce que l'équipe est capable de faire en tenant compte de sa vélocité

SCRUM, la vélocité

Lorsque l'équipe n'a jamais travaillé ensemble, alors on pourrait simuler un sprint fictif pour déterminer la vélocité de l'équipe. On pourrait également calculer sa capacité

Exemple de prévision: Trois stories sont étudiées qui avaient été estimées à 3, 2 et 5 points. Les tâches identifiées pour ces stories sont estimées à 60 heures.

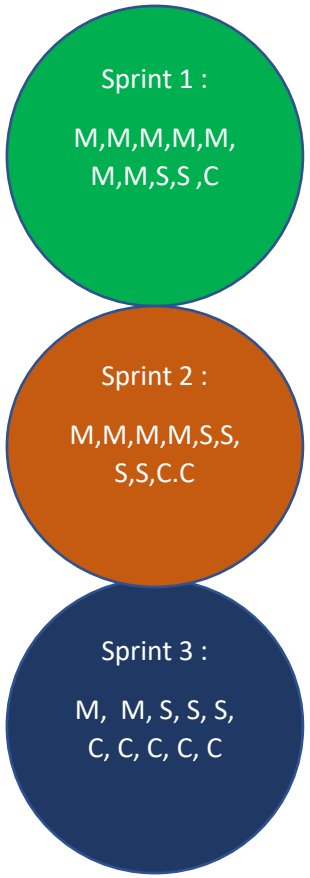
- L'équipe dispose de 300 heures pour le sprint
- En 60 heures, l'équipe pourrait réaliser 10 points. (3 +2 +5)
- En 300 heures (sprint) l'équipe pourrait réaliser (300/ 60) * 10 points.
- La capacité de l'équipe serait 50 points (estimation)

SCRUM, le plan de release

Étape 5:Produire un plan de Release

- Après avoir fait les étapes précédentes produire un plan de release devient facile, un jeu d'enfant. On procède comme suit :
  - On prend le backlog du produit priorisé et estimé.
  - On commence par le premier sprint de la release. On y associe les stories en commençant par les prioritaires
  - On continue dans ce sprint en additionnant la taille en points de stories jusqu'à atteindre la capacité de l'équipe
  - Quand on y arrive, on passe au sprint suivant.
- Et Si on ne tombe pas exactement sur la capacité de l'équipe à l'intérieur d'un sprint ?
    - On va légèrement au dessus , ou en dessous ou une story un peu moins prioritaire et qui rentre dans le sprint.
- **Garder du « lousse »**

SCRUM, le plan de release



- Dans le sprint 1, en principe on devrait retrouver plus de stories priorisées M que S ou C. Parfois un story priorisée C complète bien le sprint.
- Ce n'est pas rare, que des stories priorisée M se retrouvent dans le dernier sprint. Par contre il faut faire en sorte de les terminer en premier.
- Si on suit la méthode qui consiste à remplir le sprint d'abord par les M, et on applique la même règle aux sprint suivant, en principe votre dernier sprint ne contiendrait pas trop de stories M
- Attention: S ou C, ne veut pas dire abandonnée. (Exemple: tourner la roue pour le jeu Trivial Crack).

SCRUM, le sprint zéro

Le sprint Zéro

Le développement agile a besoin d'un sprint de départ, qui ne se termine pas nécessairement par une livraison. D'une durée variable sert à mettre le projet sur de bons rails et d'apprendre à l'équipe de travailler ensemble. Concrètement, ce que l'on doit faire durant le sprint Zéro:

- Partager une vision claire du projet
- Préparer l'environnement de développement
- Produire un backlog du produit estimé et priorisé
- Roder l'équipe sur le backlog initial
- Définir la posture ergonomique de l'interface.
- Déterminer un plan de Release.
- Selon le contexte, travailler sur l'architecture, travailler sur la BD (Conception globale)
- S'offrir une belle rétrospective.

Conclusion pour cette partie

- **Le backlog du produit** est l'élément central d'un développement agile SCRUM. Celui-ci doit-être:
  - Dûment estimé. L'estimation se fait en points.
  - Priorisé (MoSCoW)
  - Ordonné par priorité. Les M en premier
  - Les tests d'acceptation doivent être présent dans le backlog.
  - Le **PO est le responsable** du Backlog du produit. Le backlog est partagé par toute l'équipe.
  - Le backlog est unique (un seul) et il est vivant.

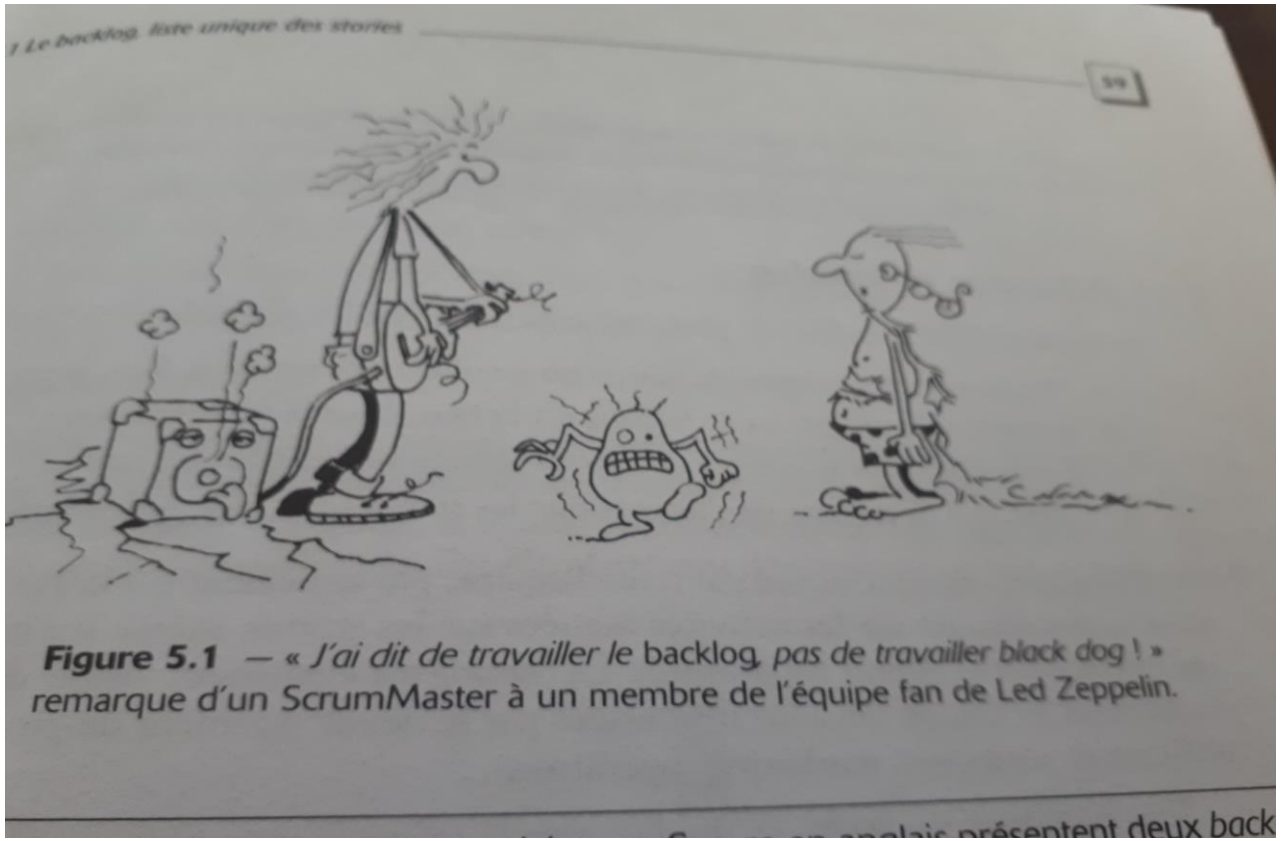


Figure 5.1 — « *J'ai dit de travailler le* backlog, *pas de travailler black dog !* » *remarque d'un ScrumMaster à un membre de l'équipe fan de Led Zeppelin.*

Source: SCRUM : le guide pratique de la méthode agile la plus populaire., Claude Aubrey, DUNOD

Conclusion pour cette partie

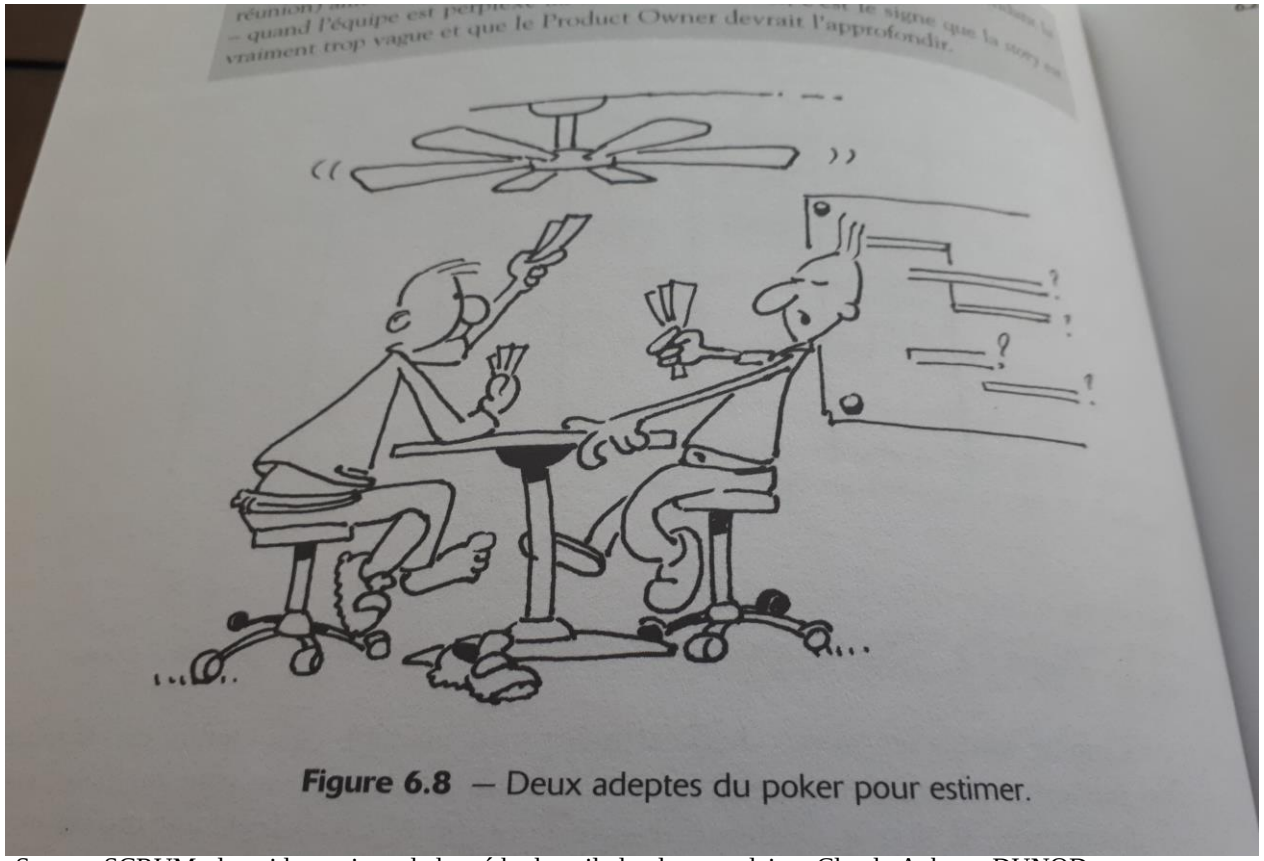


Figure 6.8 — Deux adeptes du poker pour estimer.

Source: SCRUM : le guide pratique de la méthode agile la plus populaire., Claude Aubrey, DUNOD

L'estimation des User-Stories se fait par l'équipe, en point et en utilisant le planning Poker

1.Le PO présente la story.
2.Les membre de l'équipe posent des questions pour clarifier la story.
3.Tous les participants présentent en même temps la carte choisie pour l'estimation
4.L'équipe discute des différences éventuelles entre les estimations.
5.On recommence jusqu'à une convergence des estimations.
6.On passe à la prochaine story.

Conclusion pour cette partie

Les sprints ne se chevauchent pas. Chaque sprint se termine par un produit livrable.

- Il y a des scrums quotidiens pour s'assurer que l'équipe bénéficie des meilleures conditions pour accomplir les tâches. C'est le **ScrumMaster** qui anime ces rencontres.
- Pour chaque sprint, il y a un backlog de sprint.
- Il y a une revue de sprint, à la fin du sprint.
- Il y a une rétrospective de sprint, à la fin du sprint.

La durée d'un sprint est de 3 à 4 semaines. Ne pas faire de sprints trop courts ni trop longs.

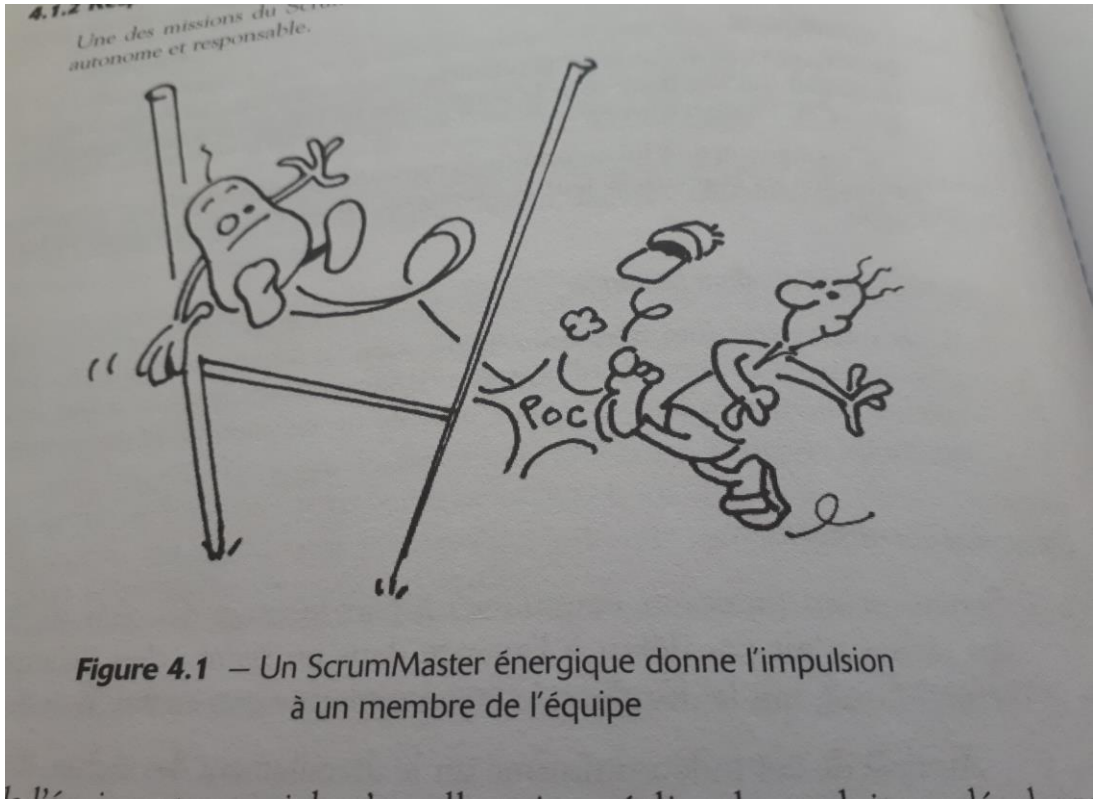


Figure 4.1 *– Un ScrumMaster énergique donne l'impulsion à un membre de l'équipe*

Source: SCRUM : le guide pratique de la méthode agile la plus populaire., Claude Aubrey, DUNOD

Conclusion pour cette partie

Une release est une version du système (logiciel). Pour sa planification, il est important de :

- De déterminer la fin de la release
- Estimer les stories
- Définir la durée du sprint
- Estimer la capacité de l'équipe (Vélocité)
- Faire le plan de release en gardant du « lousse ». Exemple, si une une équipe a 100 heures pour le sprint et que la planification est sur 90 heures alors il n'y a pas assez de lousse pour atteindre les objectif du sprint. (20% à 30 % pour le lousse)
- Les équipes expérimentées prennent du « lousse » mais ne l'estime pas en heures.

Figure 7.7 – Trop de mou dans le plan !

Source: SCRUM : le guide pratique de la méthode agile la plus populaire., Claude Aubrey, DUNOD

Sources

- Gestion de projet agile 3e Édition , Véronique Messager Rota, Eyrolles 2011.
- SCRUM : le guide pratique de la méthode agile la plus populaire., Claude Aubrey, DUNOD 2011.
- http://www.aubryconseil.com/post/Le-backlog-de-produit
- http://www.aubryconseil.com/post/2007/01/16/159-le-backlog-de-sprint
- http://www.agiliste.fr/items/bien-demarrer-avec-scrum/
- http://fr.wikipedia.org/wiki/M%C3%A9thode_MoSCoW
- https://www.nutcache.com/fr/blog/quest-ce-que-le-planning-poker-scrum/

SCRUM

QUESTIONS ??