

Méthodes agiles

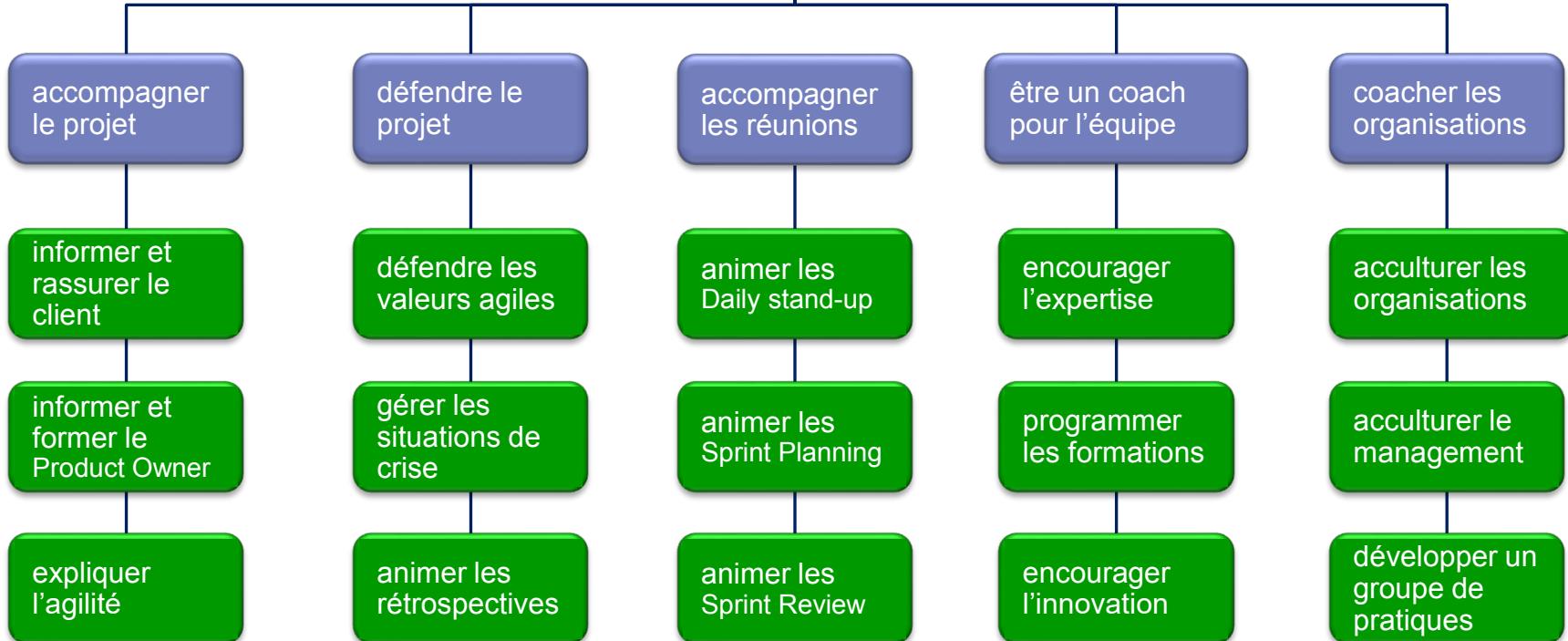
Méthode
SCRUM



Conduite de projet agile
Scrum Master



Scrum Master



Lettre de mission

Scrum
Master

Responsable de
l'adhésion à Scrum

Servant-leader au
service de l'équipe

Définit les
interactions des
partenaires

Le Scrum Master aide les personnes extérieures à comprendre quelles sont les interactions avec l'équipe Scrum qui sont utiles et celles qui ne le sont pas.

Le Scrum Master propose au Product Owner les bonnes pratiques pour la gestion du Product Backlog.

Le Scrum Master facilite les réunions et les évènements utiles au projet agile.

Le Scrum Master coache l'équipe de développement afin qu'elle s'auto-organise et fonctionne en tant qu'équipe autonome et multidisciplinaire.

Une équipe agile est une équipe multidisciplinaire composée de personnes aux compétences différentes et regroupées dans le cadre d'un projet.

Le Scrum Master agit en tant que servant-leader



1

Le Scrum Master met en place la méthode Scrum comme un processus au service de l'équipe.

2

Le Scrum Master guide l'équipe de développement afin qu'elle atteigne un niveau d'auto-organisation.

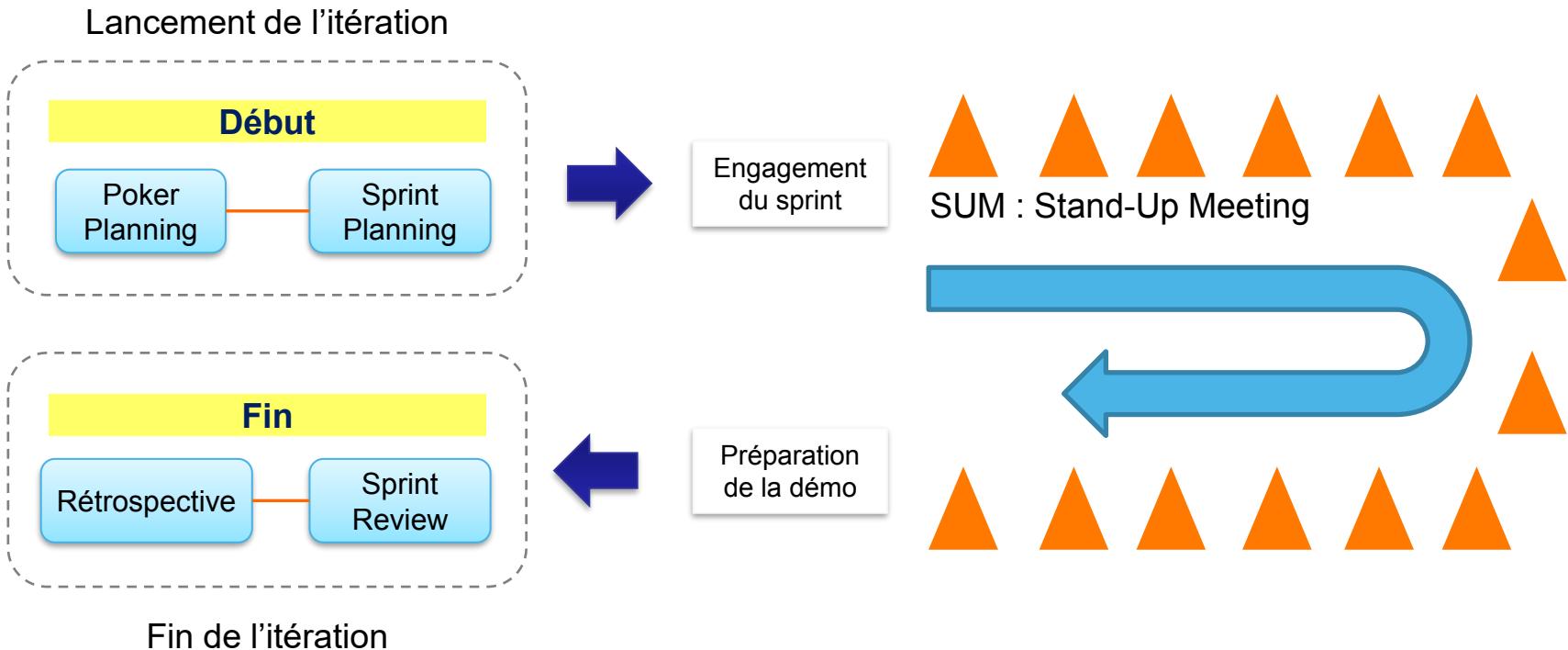
3

Le Scrum Master dirige l'équipe pendant les périodes de conflit et de débat.

4

Le Scrum Master agit en tant que coach vis-à-vis de l'équipe et en tant que référent de la méthode Scrum vis-à-vis de l'organisation.

Le Scrum Master accompagne l'équipe de développement



Le Scrum Master agit en tant que référent



1

Référent de la
méthode Scrum

2

Défense des
valeurs agiles

3

Acculturation
des organisations

4

Développement
des ressources

Le Scrum Master agit en tant que référent de la méthode Scrum en particulier vis-à-vis des principes et des valeurs agiles.

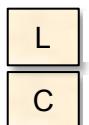
Le Scrum Master contribue aux cérémonies agiles

Scrum Master	L	C	A	N
POKER PLANNING		X		
SPRINT PLANNING	X			
DAILY STAND-UP	X			
SPRINT REVIEW	X			
RETROSPECTIVE	X			

Le Scrum Master accompagne et/ou dirige les cérémonies agiles.

Le Scrum Master accompagne un Product Owner débutant lors de sa participation aux premières réunions agiles.

Légende



Leader



Auditeur



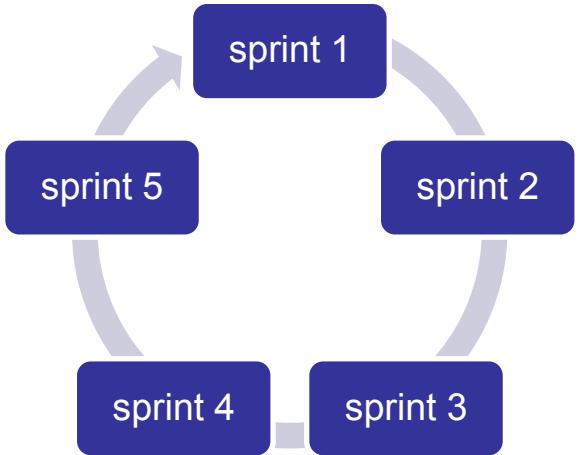
Contributeur



Ne participe pas

Les méthodes agiles

Merci



Méthodes agiles

Méthode
SCRUM



Conduite de projet Agile
Présentation des cérémonies



Les cérémonies agiles

La conduite régulière des cérémonies agiles assure le bon fonctionnement de la méthode Scrum.

Sprint Planning : une réunion de lancement avant le début de l'itération.

Daily Stand Up : une réunion quotidienne de suivi avec les développeurs.

Sprint Review : une réunion de validation des travaux à la fin de l'itération.

Début de l'itération

Sprint Planning

Sprint Review

Fin de l'itération

La réunion de lancement Sprint Planning inclut une réunion d'évaluation (appelée Poker Planning) de la complexité des stories de la prochaine itération.

Daily Stand-Up

La réunion Sprint Review peut être suivie d'une réunion appelée Rétrospective visant à améliorer la qualité et l'efficacité des prochaines itérations.

Les cérémonies agiles

Décomposition
des besoins et
des exigences

Priorisation

- Valeur métier
- Connaissance
- Niveau d'effort
- Risques

Alimentation
continue du
backlog

Product Backlog

User Stories



Evaluation
Engagement
Démarrage

Poker Planning

Sprint Planning

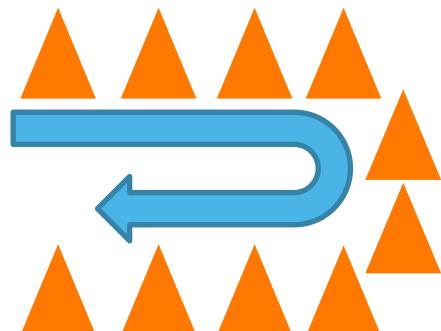
Fin de l'itération

Rétrospective

Sprint Review



Daily Stand-Up



Daily Stand-Up

Les cérémonies agiles

Evaluation
des stories Lancement
de l'itération

Poker
Planning

Sprint
Planning

Rétrospective

Sprint
Review

Fin de l'itération

Travaux de
développement

Daily
Stand Up



Le Scrum Master dirige et/ou accompagne les cérémonies agiles.



Les cérémonies agiles accompagnent la vie des projets agiles. Le Scrum Master est le référent de la méthode Scrum. Il intervient en particulier dans la conduite des réunions au bénéfice des projets agiles.

Les cérémonies agiles

Evaluation par les DEV des stories présentées par le Product Owner.

Planning des tâches de chaque membre réalisé par le Scrum Master.

Poker Planning

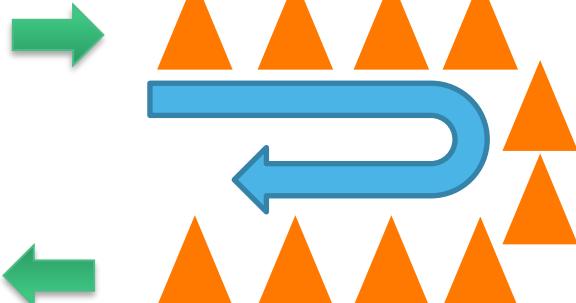
Sprint Planning

Rétrospective

Sprint Review

Identifier ce qui a bien fonctionné et ce qui a moins bien fonctionné pendant l'itération..

Validation par le Product Owner des stories réalisées par les développeurs.



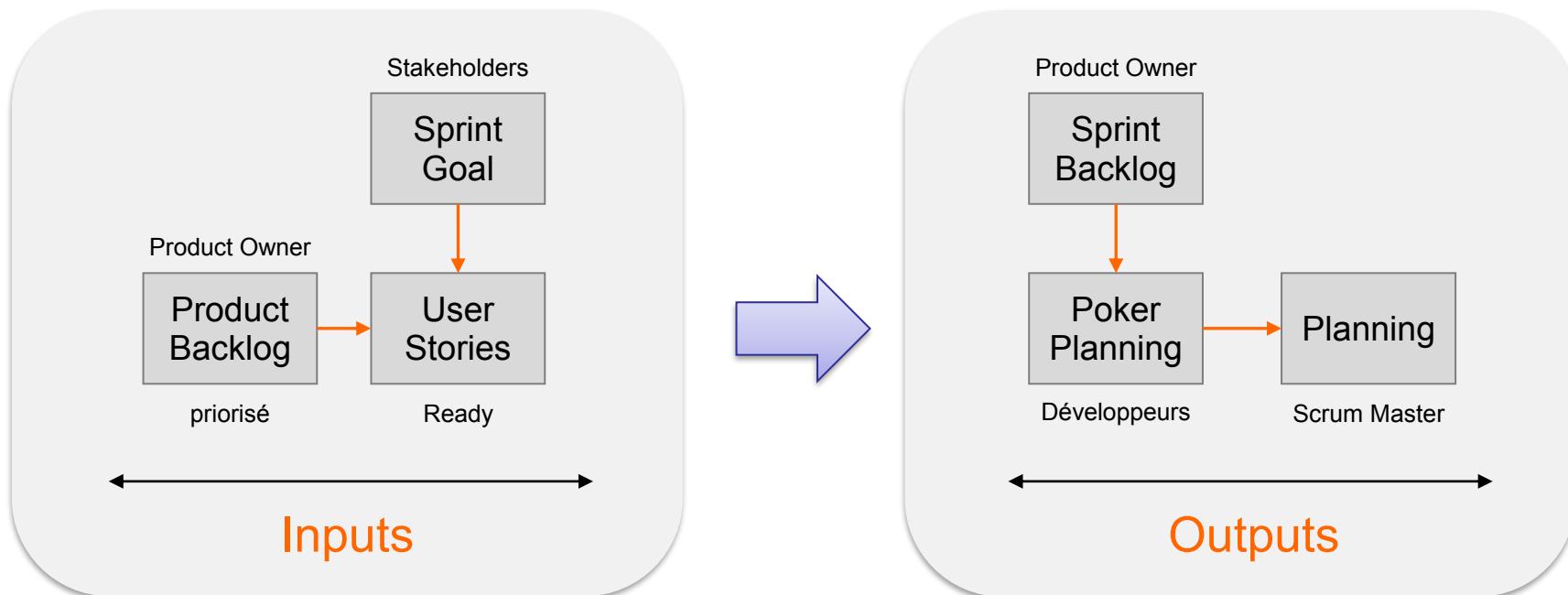
Point quotidien de suivi des travaux et identification des points de blocage.

Les cérémonies agiles contribuent au bon fonctionnement des projets agiles.



Les cérémonies agiles sont indispensables au fonctionnement régulier des itérations.

Prérequis et livrables



Poker Planning : technique d'estimation agile

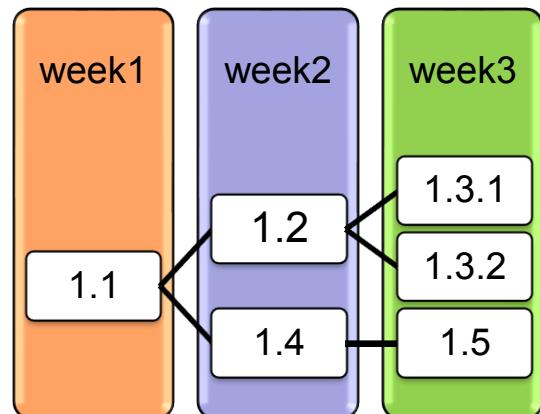
Planning : définit les tâches dont chaque développeur a la charge durant l'itération

Comment ça marche ?

- ❑ Le Sprint Planning fait suite à la réunion d'évaluation (Poker Planning) avec la liste des stories engagées.
- ❑ Le Scrum Master prend en charge le découpage de chaque story de l'itération en tâches.
- ❑ Le découpage des stories (y compris les études) permet la construction du planning de l'itération et l'affectation des tâches aux développeurs.
- ❑ Le Sprint Planning peut prendre 1 ou 2 heures.

A noter : lors du Poker Planning les développeurs ne savent pas quel développeur prendra en charge quelle story. Cette information n'est disponible qu'à l'issue du Sprint Planning.

Itération en cours



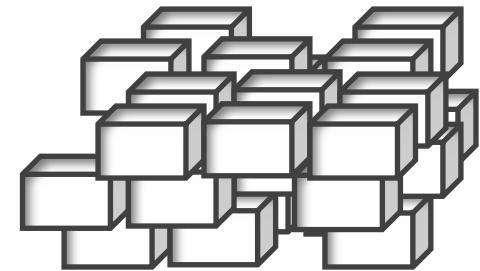
Planning des développeurs

Comment ça marche ?

1

- Les Stories priorisées par le Product Owner pour la prochaine itération sont extraites du Product Backlog.
- Le Product Owner présente chaque Story à l'équipe de développement.
- Le Product Owner répond aux questions des développeurs.
- Le Poker Planning est un moment d'échange et de dialogue entre le PO et les développeurs.

User Stories READY



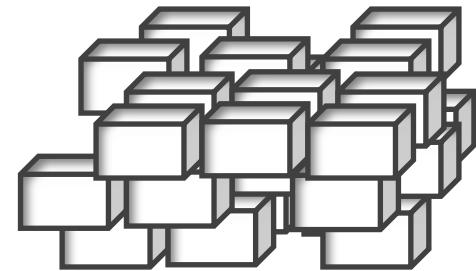
A noter : lors du Poker Planning le Product Owner présente uniquement des stories priorisées qui sont conformes au mémo READY et au mémo INVEST.

Comment ça marche ?

2

- ❑ Chaque développeur évalue le niveau d'effort (appelé complexité et exprimé en points) de la Story et s'exprime par un vote.
- ❑ Les développeurs ne s'influencent pas : pour ce faire ils votent tous en même temps.
- ❑ Le vote est exprimé en points de complexité : les développeurs utilisent les nombres de la série de Fibonacci (1, 2, 3, 5, 8, 13, 21, ∞).
- ❑ Chaque développeur utilise un jeu de cartes dont chaque carte porte un nombre de la série de Fibonacci.

User Stories READY



(*) série de Fibonacci : série dont chaque terme est la somme des deux termes consécutifs précédents

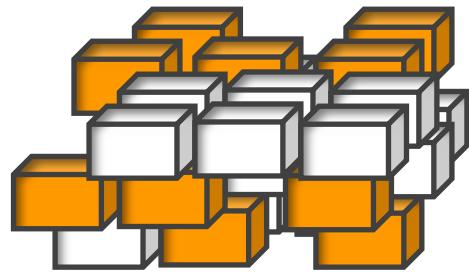
Comment ça marche ?

3

A la fin du premier vote :

- ❑ le vote qui exprime la complexité la plus élevée est questionné afin d'expliquer les raisons et les risques qui motivent ce vote.
- ❑ le vote qui exprime la complexité la plus faible est questionné afin d'expliquer les raisons et/ou l'absence de risques qui motivent ce vote.
- ❑ le Poker Planning est un moment d'échange et de dialogue entre les développeurs.

User Stories évaluées

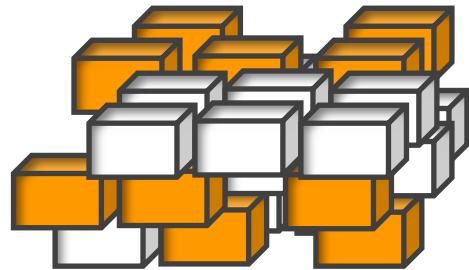


Comment ça marche ?

4

- ❑ Chaque développeur pose des questions et reçoit les réponses utiles afin de réviser le cas échéant son vote.
- ❑ Un deuxième vote est engagé afin d'obtenir un consensus de l'ensemble des développeurs : la valeur finale de la complexité de la Story est alors enregistrée.
- ❑ Chaque Story est évaluée selon le même processus.

User Stories évaluées

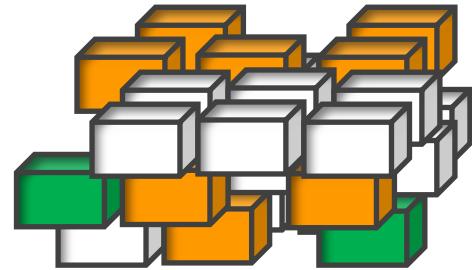


Comment ça marche ?

5

- ❑ Lorsque l'ensemble des Stories est évalué, la somme des votes est comparée avec la vélocité de l'équipe de réalisation.
- ❑ La vélocité représente la capacité à faire de l'équipe des développeurs pendant une itération.
- ❑ Si la somme des votes excède la capacité à faire, alors le nombre des Stories de la prochaine itération est réduit par le PO afin de demeurer dans la limite de la capacité des développeurs.
- ❑ Le Product Owner décide du contenu de l'itération à venir dans l'intérêt des utilisateurs.

User Stories évaluées



8 User Stories engagées

Légende



READY à évaluer



READY évaluée



READY dépriorisée

Ce qu'il faut retenir

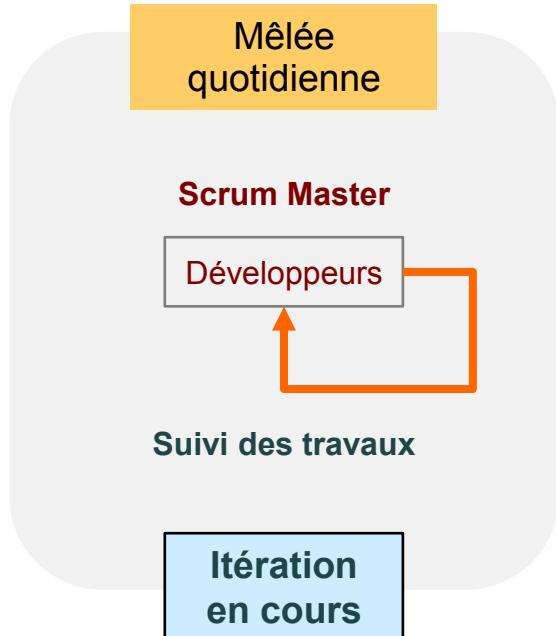
6

Le Poker Planning est une technique d'estimation agile :

- Tous les développeurs participent.
- Le vote est individuel.
- Pas d'influence entre les développeurs.
- Le Poker Planning génère du dialogue entre les acteurs.
- Le Poker Planning améliore la connaissance du projet.
- Le Poker Planning fonctionne parce que c'est fun !

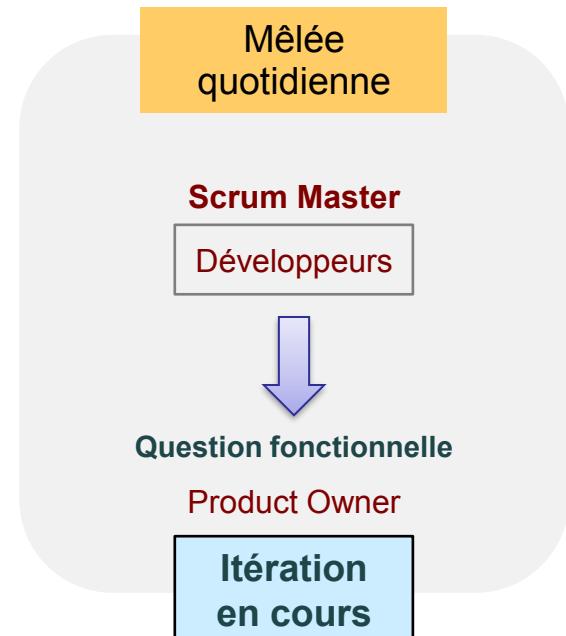
Comment ça marche ?

- Une réunion quotidienne animée par le Scrum Master afin de faire le point sur les travaux du sprint en cours.
- Une réunion tous les jours : même heure, même endroit, 15' minutes maximum.
- Chaque développeur expose ses travaux, ses questions et ses points de blocage :
 - ▶ Qu'ai-je fait hier ?
 - ▶ Que vais-je faire aujourd'hui ?
 - ▶ Quels sont les problèmes qui me ralentissent ?
 - ▶ Quelles sont les questions qui demeurent sans réponse ?



Comment ça marche ?

- Si le Product Owner est présent, il écoute mais ne prend pas la parole lors de la réunion quotidienne.
- Si le Product Owner est absent, il doit prévoir de se rendre disponible après la réunion.
- En effet, le Product Owner doit apporter rapidement une réponse aux questions fonctionnelles qui lui sont adressées et qui peuvent ralentir ou bloquer les développeurs.

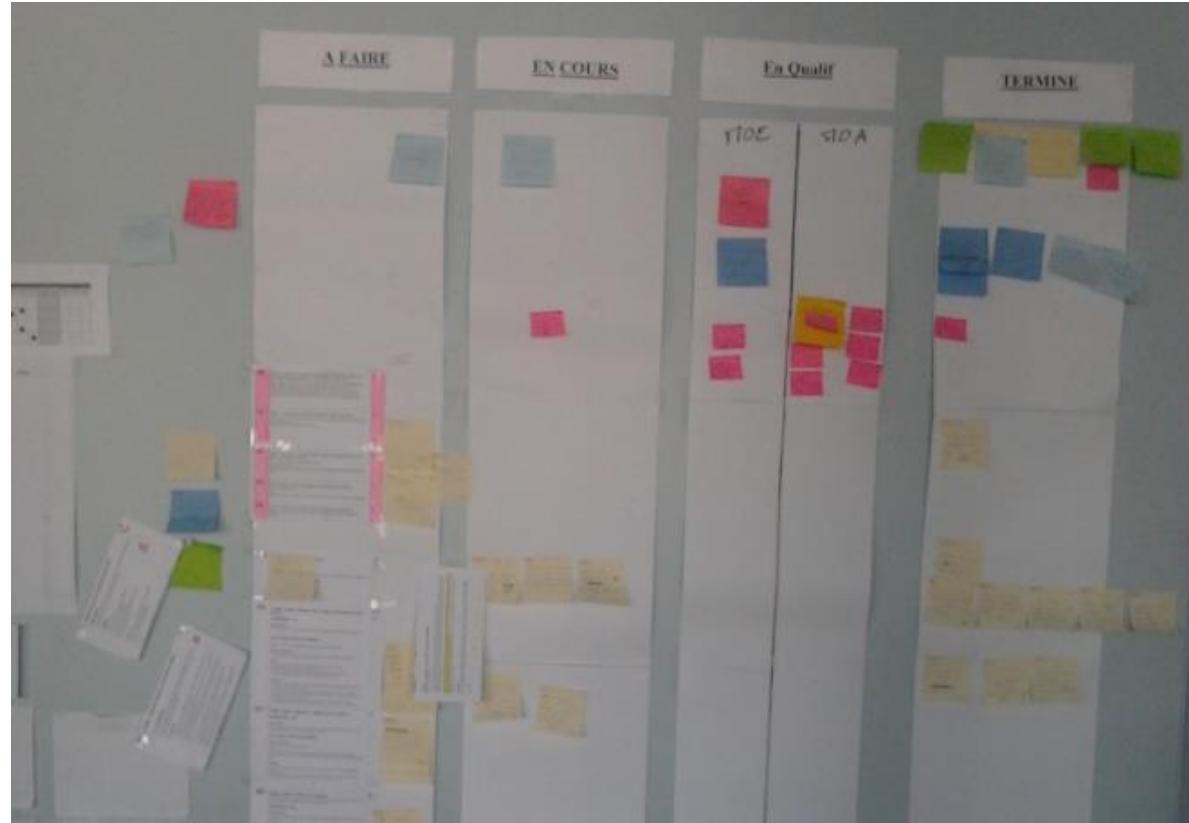


Comment ça marche ?

Un exemple de suivi des travaux d'une itération.



Board-Sprint26.jpg



Sprint
Review

Validation des stories du Sprint

1



Sprint N+1

Validation au fil de l'eau

Bonne pratique

2



Sprint N+1

Validation en fin de sprint

Alternative

Sprint Review

Validation des stories du Sprint

- Les travaux du Sprint sont disponibles pour une présentation.
- Chaque Story est présentée:
 - ▶ mise en œuvre du rôle embarqué dans la User Story
 - ▶ vérification des règles de gestion
 - ▶ vérification de l'interface utilisateur (IHM)
 - ▶ validation des résultats attendus
- Le Product Owner accepte/refuse les Stories réalisées pendant l'itération :
 - Les Stories validées sont éligibles pour une prochaine livraison (Go Live) au bénéfice des utilisateurs ; elles disparaissent de facto du Product Backlog ; elles sont archivées.
 - Les Stories refusées seront embarquées dans la prochaine itération avec la même complexité.

RAPPEL

Mesure de la performance du Done

SPRINT [1,N]

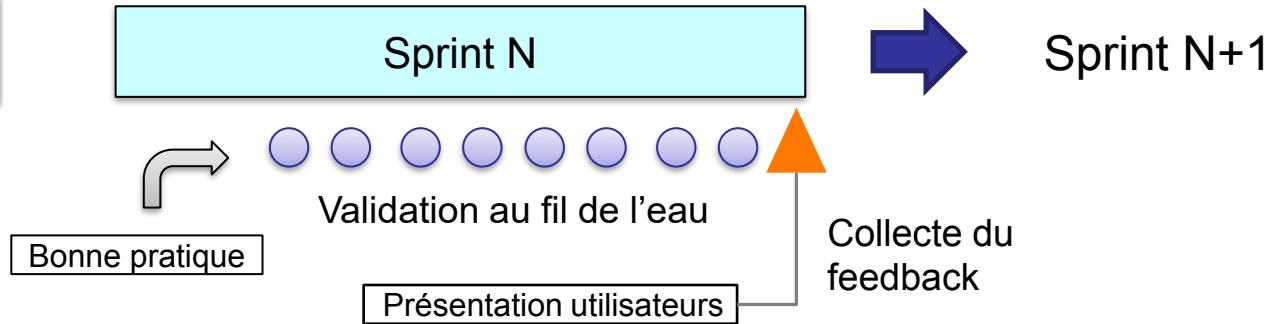
Σ (nombre (stories Done))

Σ (nombre (stories))

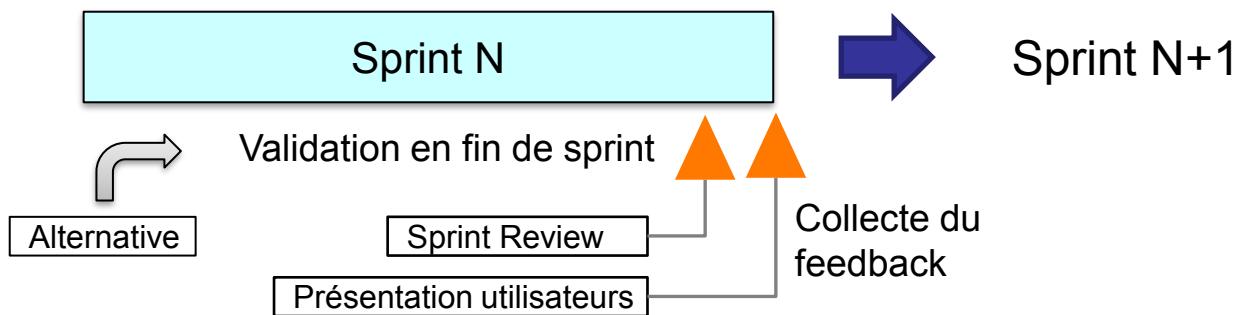
Sprint
Review

Présentation aux utilisateurs

1



2



Présentation aux utilisateurs



Les utilisateurs sont légitimes pour être présents lors de la validation des stories (vérification des critères du Done).



Si les utilisateurs montrent peu d'intérêt pour la validation des stories (critères du Done), le retour d'expérience montre qu'ils sont intéressés par une présentation des stories qui sont démontrables.

La bonne pratique est que les utilisateurs réservent leurs questions et leurs demandes pour la fin de la réunion afin que le dialogue entre les développeurs et le Product Owner demeure efficace et concentré sur les stories à valider.

La bonne pratique est que la collecte du feedback des utilisateurs soit réalisée lors de la présentation des stories qui sont démontrables.

Comment ça marche ?

- La rétrospective s'inscrit dans une démarche Qualité d'amélioration continue.
- Chaque membre de la communauté (Client, Scrum Master, Product Owner, développeurs, représentants métier, représentants utilisateur, etc.) participe à la rétrospective.
- Il s'agit d'identifier ce qui a bien fonctionné et ce qui a moins bien fonctionné dans le processus de l'itération :
 - ▶ chacun vote pour 2 items qui se sont bien passés
 - ▶ chacun vote pour 2 items qui se sont mal passés

RETROSPECTIVE

Démarche Qualité
d'amélioration continue



Utilisation des Post-It

Comment ça marche ?

- Les items qui se sont mal passés sont classés en fonction du nombre total de votes obtenus :
 - ▶ les 2 items qui ont reçu le plus de votes sont identifiés,
 - ▶ une action d'amélioration est définie pour chacun de ces 2 items,
 - ▶ le suivi des actions sera réalisé lors des prochaines rétrospectives jusqu'à ce que le processus d'amélioration soit déclaré satisfaisant

RETROSPECTIVE

Démarche Qualité
d'amélioration continue



Utilisation des Post-It

Rétrospective

Comment ça marche ?

Démarche Qualité
d'amélioration continue

Un exemple de rétrospective à l'issue d'une itération.



Microsoft
PowerPoint Presentat

Liste des items qui se sont bien passés



Liste des items qui se sont mal passés

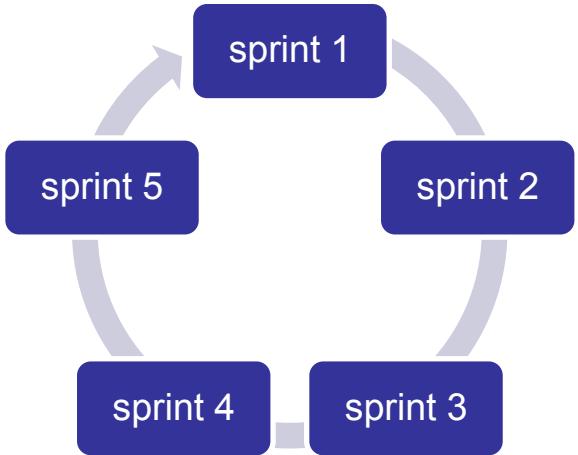


a. Intervention du coach pendant Démo/Poker (avis externe + recadrage « façon de faire »)	a. 20 – Périmètre / Formalisation des US non figé avant Poker Planning => impact sur chiffrage
b. Déroulement Démo : doc. d'une démo technique fort appréciée + moins de digressions	b. 9 - Mauvais déroulement du Poker Planning (il faut avoir une seule idée à débattre)
c. Beaucoup plus d'échanges MOA/MOE	c. 2 - RTG à « l'arrache » et peu productif (setup mal organisé; rappel : RTG décalé vendredi après-midi au lieu du vendredi matin)
d. Réel effort d'explication des TS par MOE (langage simple et compréhensible pour les représentants métier)	d. 14 – Problème planification RTG vis-à-vis Démo
e. Projet arrivé « enfin » en vitesse de croisière (car beaucoup moins d'anomalies (25 => 3 !))	e. 1 - Visibilité/capacité de réalisation/vélocité mal partagée entre MOA/MOE en vue de la définition du Sprint.
f. Interaction avec le nouveau chef de projet Exploitation = confiance.	f. 13 - QC et Mingle non mis à jour = incohérence (constat partagé MOE/MOA)
	g. 1 - cette rétrospective est effectuée post Poker
	g. 5 - Mauvaise qualification

Description des actions en cours	Status
<p>1 - PERIMETRE / FORMALISATION US :</p> <p>a. Définition par le Product Owner d'un statut US « Ready » : → un porteur + US comprise par tous, chiffrable et testable.</p> <p>b. Démarche :</p> <ul style="list-style-type: none"> - Co-construction US lors club PO, - Améliorer les échanges MOE/MOA SI « au fil de l'eau » : <ul style="list-style-type: none"> o afin de valider le périmètre des US (construction/découpage en tenant en compte des impératifs technique), o via point de passage téléphonique « faisabilité ». 	<p>Action Ouverte</p> <p>Les points a) et b) ont été initialisés</p> <p>Efforts à continuer.</p>
<p>2 – PLANIFICATION ORGANISATION RTG</p> <p>Définir une Roadmap et la partager entre MOE/MOA SI (action PO).</p>	<p>Action Close</p> <p>Roadmap affichée sur le Board MOE et MOA</p> <p>Veiller à ne pas planifier un RTG la même semaine qu'une Démo.</p>
<p>3 - Planning des Versions / Release / Sprints basé sur le Product Backlog à partager avec l'équipe MOE (manque de visibilité)</p>	<p>Action Ouverte</p>
<p>4 - Manque d'anticipation des US par sprint :</p> <p>→ Volume OK mais priorisation KO</p> <p>a. Point à améliorer en club PO</p> <p>b. Affichage PPT sur le Board MOE et sur le Board MOA</p>	<p>Action Ouverte</p>

Méthodes agiles

Merci

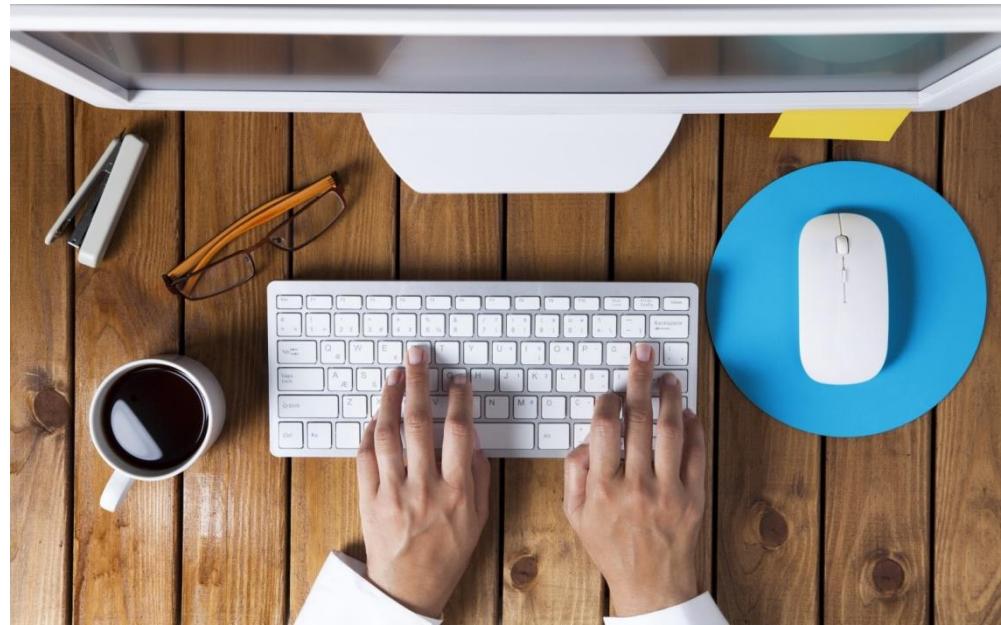


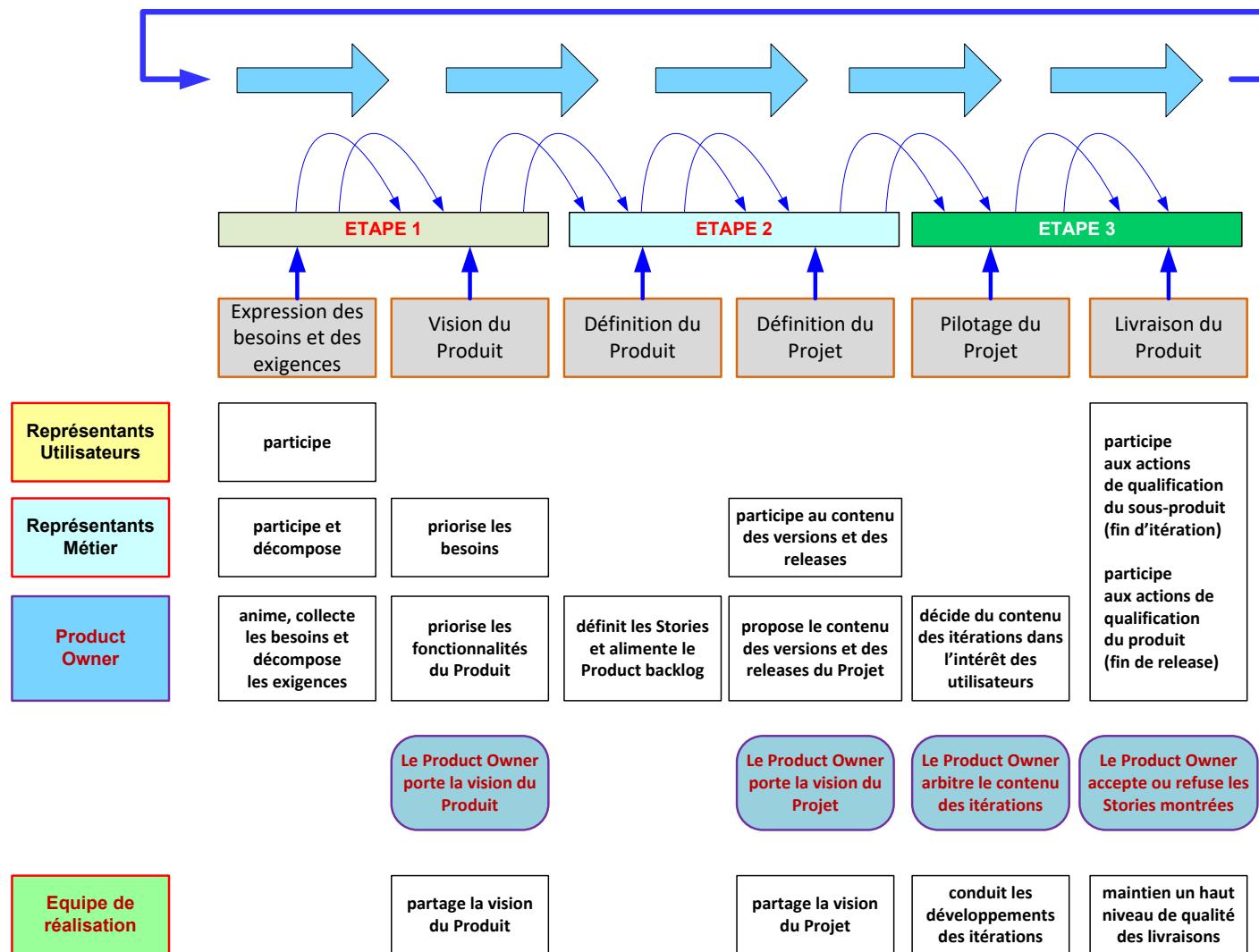
Méthodes agiles

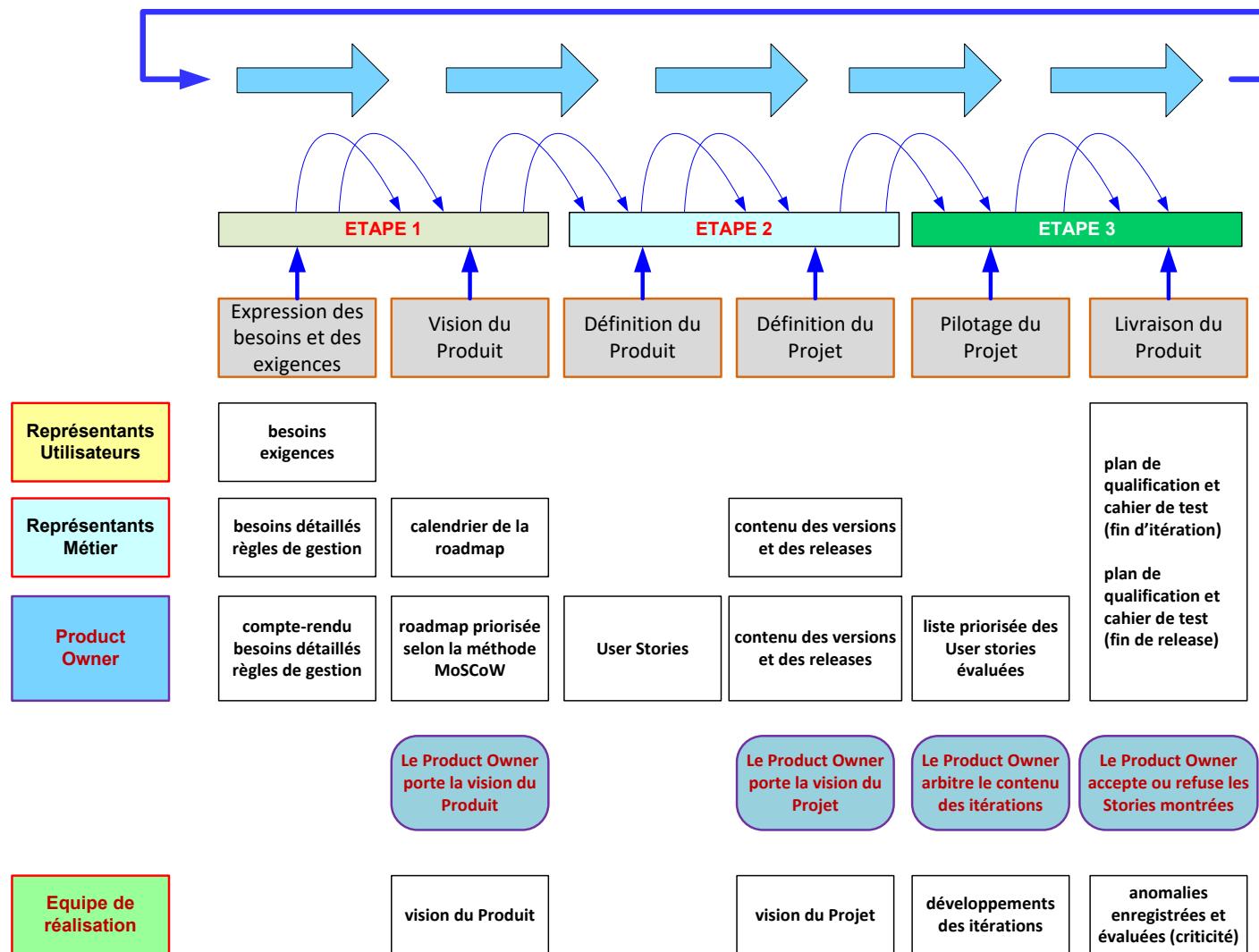
Méthode
SCRUM



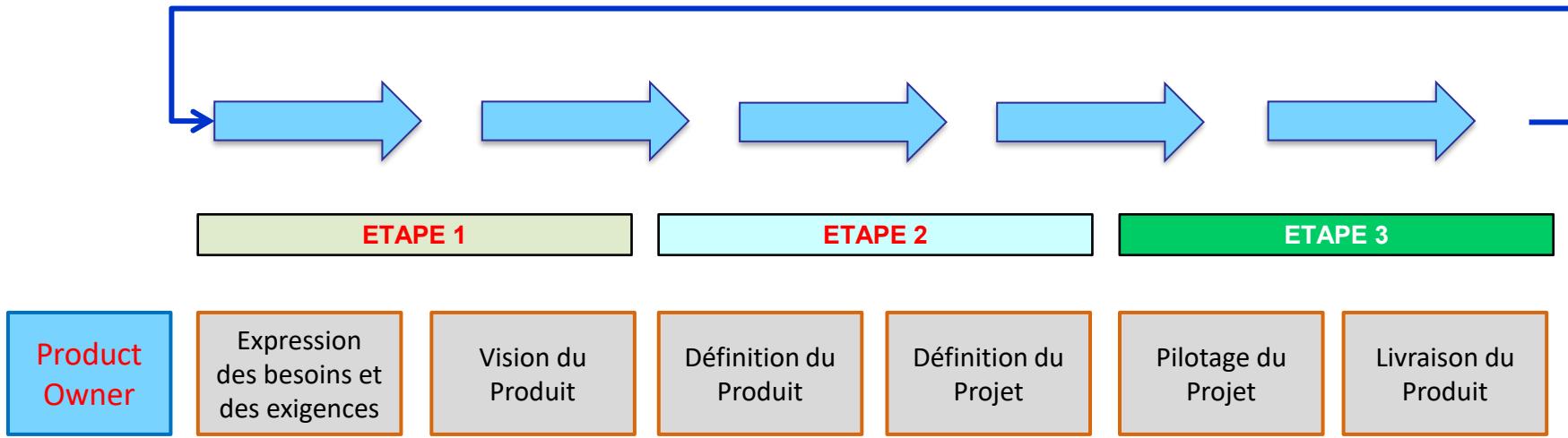
Conduite de projet agile
Les activités du projet







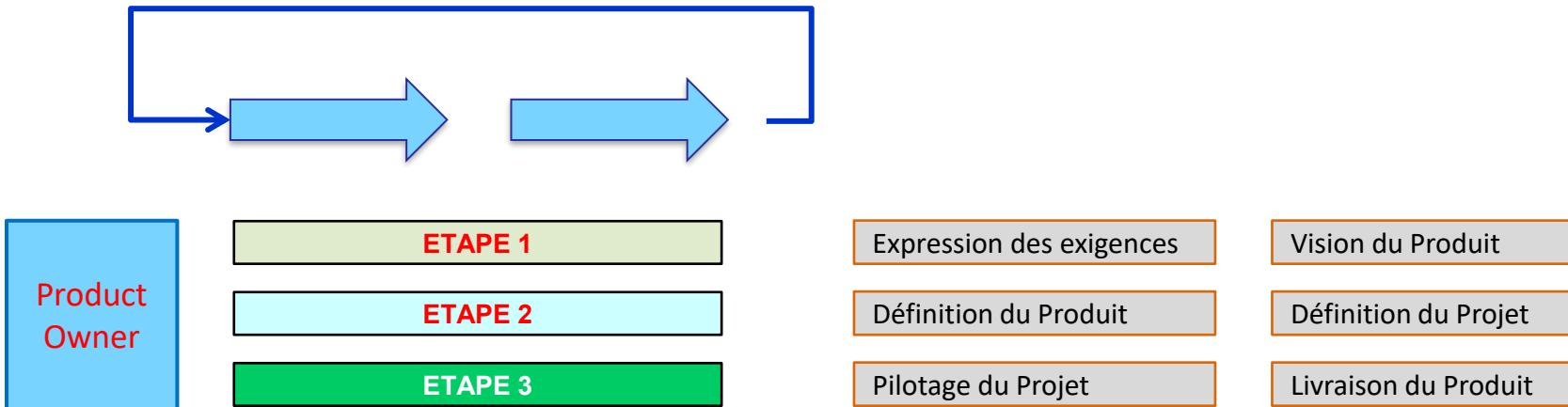
Les activités du Product Owner



Au départ, les 3 plans de travail du Product Owner s'ouvrent successivement dans le temps. En effet, au début du projet le Product Backlog est vide et la collecte des besoins doit être engagée en tout premier lieu.

L'étape 3 signifie que la décision d'affecter une équipe de développeurs au projet a été prise par le client compte tenu du niveau d'alimentation du Product Backlog avec des stories READY.

Les activités du Product Owner

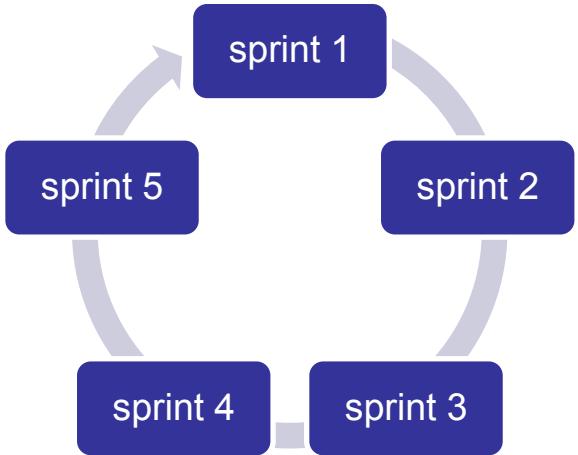


A partir du moment où l'équipe de développeurs est active, les 3 plans de travail du Product Owner fonctionnent en parallèle. Bien entendu, la disponibilité à 100% du Product Owner est indispensable pour réussir un projet agile compte tenu de tous les travaux à conduire.

A noter que la fonction de Product Owner peut être tenue par plusieurs personnes. Ce qui explique le terme "Armée de PO" que l'on peut trouver dans la littérature agile.

Méthodes agiles

Merci



Méthodes agiles

Méthode
SCRUM



Conduite de projet agile
Les outils du projet



Les outils

Vue globale

Gestion du
Product Backlog

Vélocité

Vue de détail

Rédaction des
user stories

Rally Test Game

Décomposition
des besoins

Organisation du
plan de livraison

Bateau agile
Speed Boat

Mission

découvrir les rôles

décomposer les exigences

définir le produit

organiser le produit

valider le produit

Outil

Personas

Arbre de décomposition

Product Backlog

Product Backlog

Critères du DONE

Livrables

ROLES

EPIC

STORIES

FEATURE

Mémo INVEST

VERSION

RELEASE

Mémo READY

SPRINT

Mini tutoriels



ETAPE 1

ETAPE 2

ETAPE 3

Besoins et exigences

Vision du Produit

Définition du Produit

Définition du Projet

Pilotage du Projet

Livraison du Produit

Collecte des besoins

Vision

Alimentation Product Backlog

Priorisation

Cérémonies

Planification

- Brainstorming
- Atelier
- Workshop
- Interview
- Mind map
- Story map

- Versions
- Releases
- Sprints
- Stories

- User stories
- Technical stories
- Anomalies
- Etudes

- Valeur métier
- Connaissance
- Niveau d'effort
- Risques

- Sprint Planning
- Poker Planning
- Daily Stand-Up
- Sprint Review
- Retrospective

- Planification à date fixe
- Planification à périmètre fixe

- Personas
- Rôles

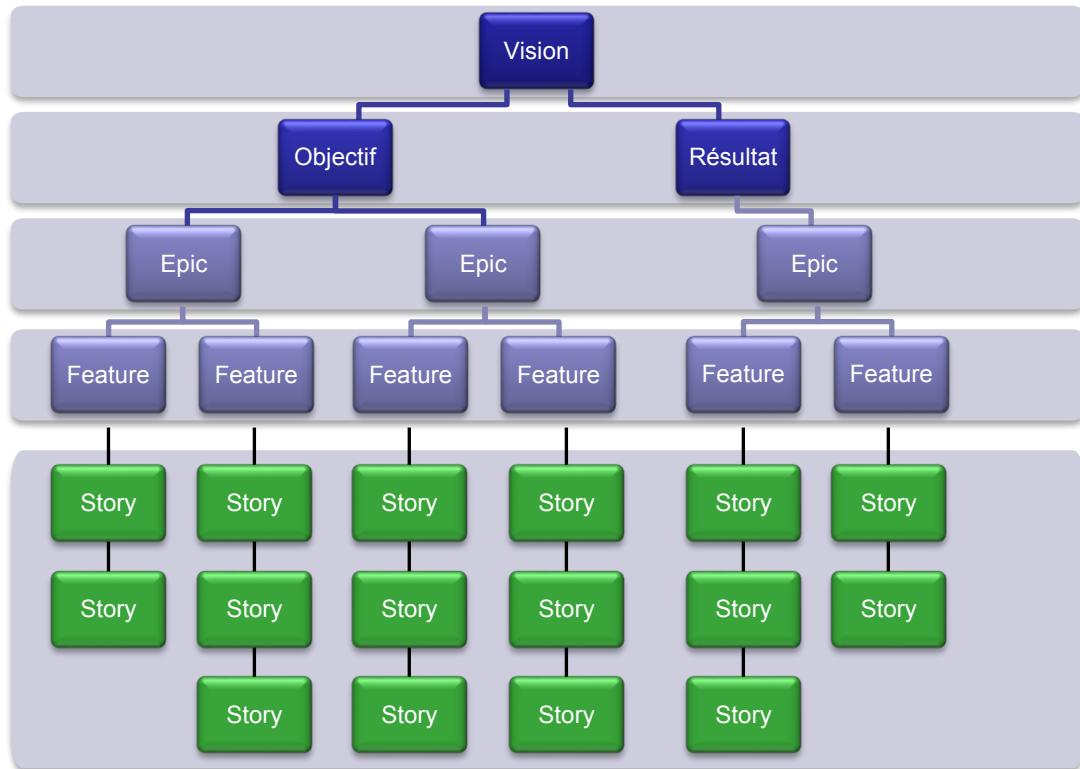
- enthusiasts
- visionaries
- pragmatists
- conservatives

- EPIC
- Features

- MUST
- SHOULD
- COULD
- WON'T



Décomposition des besoins et des exigences



Arbre de décomposition

Plusieurs techniques (*) sont utilisables pour collecter les exigences d'un produit :

- Brainstorming
- Atelier / Workshop
- Interviews
- Mind Map
- Story Map

(*) Ces techniques ne sont pas spécifiquement Agile.



Décomposition des besoins et des exigences

Objectif

Epic

Ensemble trop large pour ne pas être décomposé ou pour le niveau d'effort d'une itération.

Feature

Capacité du produit qui requiert plusieurs interactions des rôles utilisateurs et inclut un niveau de complexité élevé.

Story

Story liée à un seul rôle
Story de faible complexité

Les besoins et les exigences sont définies étape par étape, en partant de la vision du Produit jusqu'à la rédaction précise des Stories :

- A chaque niveau, le besoin ou l'exigence est collecté et décomposé,
- Il est possible de découper un besoin ou une exigence sur un même niveau,
- Le nom du niveau (Epic, Feature ...) n'est pas si important que cela,
- **A la fin, il n'y a plus que des Stories.**



Décomposition des besoins et des exigences

POURQUOI ?

Obtenir un Product Backlog facile à évaluer et à prioriser.

Décomposer les objectifs exprimés par le client et les résultats attendus du Produit.

Maintenir et valoriser le savoir-faire du client et celui de l'entreprise.

COMMENT ?

Décomposer les besoins et les exigences.

Approfondir le niveau de connaissance des besoins et des exigences.

Identifier et décrire les non-dits et les informations implicites.

QUOI ?

Livrer aux développeurs des éléments simples et de petite taille à évaluer.

Livrer des IHM et des règles de gestion détaillées et validées par le métier.

Livrer aux développeurs une information complète avec une documentation à jour.



Définir vos objectifs

1

Le contenu du Product Backlog reflète la bonne santé d'un projet Agile.

2

Le Product Backlog est la colonne vertébrale du projet Agile.
Pas de Product Backlog = pas de projet

3

Alimenter le Product Backlog est donc indispensable à la vie d'un projet Agile.

4

Il s'agit d'une activité qui doit être réalisée par le Product Owner pendant toute la durée du projet agile.



Réaliser vos objectifs

(*) Cette réserve est nécessaire afin d'autoriser les absences du Product Owner : congés, maladie, formation, etc.

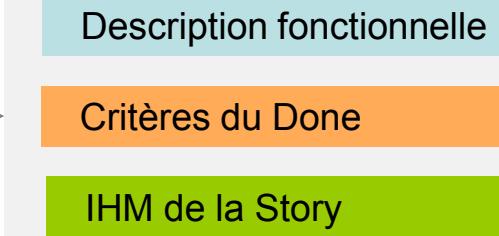
- 1 Le Product Backlog est alimenté grâce à la décomposition des besoins et des exigences.
- 2 Les Stories doivent être de qualité.
Consulter les mémos INVEST et READY.
- 3 Les Stories doivent être évaluées par l'équipe de réalisation avec un niveau d'effort exprimé en points de complexité.
- 4 Les Stories doivent être priorisées afin que la partie haute du Product Backlog alimente les prochaines itérations.
- 5 Les Stories doivent être disponibles en nombre suffisant pour que l'équipe de réalisation puisse travailler pendant 2 itérations sans compter l'itération en cours (*).



User Stories : une façon simple d'exprimer un besoin du produit

- 1 Les 3 parties d'une User story se complètent.
Elles ne doivent pas se contredire.
- 2 Pensez à des User stories supplémentaires afin de prendre en compte les affichages (listes, tableaux, etc.).

Exemple :
1- User story pour le moteur de recherche
2- User story pour l'affichage des résultats de la recherche
- 3 Vérifiez que les actions de l'utilisateur décrites dans les Critères du Done peuvent être réalisées à l'aide de l'IHM de la Story.



User Stories : une façon simple d'exprimer un besoin du produit

1

Description fonctionnelle

- Un **Titre**
- En tant que **<rôle utilisateur>**
- Je veux **<quelque chose>**
- Pour que **<une raison>**
- Selon **<les règles de gestion suivantes>**
- Tout autre élément qui donne de l'information utile (schéma, diagramme d'état, notes, ergonomie, etc.)

2

Critères du Done

- Des critères d'acceptation doivent être ajoutés à chaque User Story.
- Ces critères doivent être présentés sous la forme de **tests**.
- La rédaction des critères du Done correspond à celle d'un mini tutoriel incluant les instructions d'exécution pas-à-pas du rôle utilisateur pour chaque test.



User Stories : une façon simple d'exprimer un besoin du produit

1

Description fonctionnelle

Introduisez les données utiles à la story. Décrivez les règles de gestion qui les unissent. Relisez cette partie de la story lorsque vous aurez renseigner les Critères du Done. Il s'agit ici de vérifier la cohérence entre les règles déclarées et les règles utilisées.

2

Critères du Done

Mettez-vous dans les chaussons du rôle utilisateur afin d'utiliser son langage lors de la rédaction des cas de test sous la forme d'un mini tutoriel.

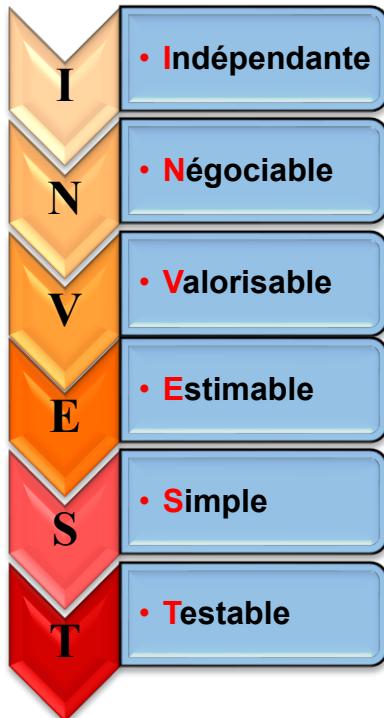
3

Listes et tableaux

Pensez à rédiger les stories utiles aux affichages. En effet, les listes et les tableaux contribuent à la valeur métier. Les rôles utilisateur vous remercieront car les listes et les tableaux portent une grande partie de l'information qui aide le rôle utilisateur à maîtriser les données courantes de son environnement métier et donc à décider quel sera son prochain clic. Les listes et les tableaux contribuent donc à leur efficacité.



User Stories : Mémo INVEST



1. Indépendante des autres User Stories et de la technologie
2. Promesse de discussion entre l'équipe MOE et le Product Owner
3. Source de valeur ajoutée pour l'utilisateur
4. L'équipe est en mesure d'estimer l'effort de réalisation
5. Suffisamment petite pour être implémentée en 1 seule itération
6. Peut être formellement validée pour être acceptée



User Stories : Mémo READY

L'équipe de réalisation ne doit investir que dans une Story qui est READY.



1. La contribution à la vision est claire
2. La valeur métier est connue
3. Le livrable est défini
4. Les critères d'acceptation sont décrits
5. Le niveau d'effort (en points) est estimé
6. La story est petite
7. L'équipe de réalisation dit au PO : « Ok, on voit ce que tu veux »



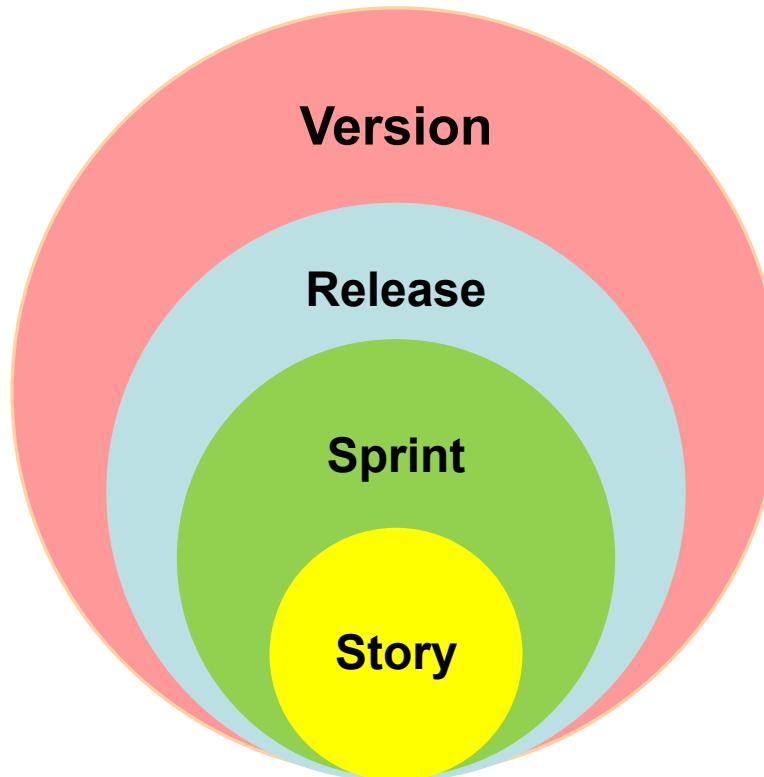
Organisation du plan de livraison des fonctions du Produit

Version

Quels sont les objectifs métier que la version doit satisfaire ?

Release

Quelles sont les fonctionnalités importantes qui sont attendues par les utilisateurs ?



Sprint

De quelle manière cette itération nous rapproche-t-elle des objectifs de la Vision du client ?

Story

A quel rôle utilisateur cette story s'adresse-t-elle ?

Quel est son comportement ?
Quelle est sa présentation ?

Comment vais-je décider si elle est terminée (Done) ?



La vélocité de l'équipe de réalisation

- 1** La vélocité est une mesure de vitesse.

- 2** La vélocité indique la quantité de Product Backlog qui peut être traitée à chaque itération.

- 3** La vélocité dépend du nombre de développeurs et de leur expérience.

- 4** La vélocité dépend de la durée des itérations.

Tips

La vélocité est utilisée par le Product Owner lors des exercices de planification.



La vélocité de l'équipe de réalisation

Comment ça marche ?

1

Une tâche de référence est choisie par l'équipe de réalisation afin de représenter 1 point.

2

Les stories du projet seront évaluées à partir de cette tâche de référence lors de chaque Poker Planning.

3

La vélocité d'une équipe de réalisation est déterminée après les 2 ou 3 premières itérations du projet.

4

La vélocité de l'équipe de réalisation devient alors une constante qui correspondra à sa capacité à faire.

Tips

Chaque équipe agile fait l'apprentissage de sa propre vélocité.



La vitesse de l'équipe de réalisation

Comment ça marche ?

1

Une modification par le projet agile de la durée de ses itérations a un impact sur la vitesse.

2

Un changement dans l'équipe (arrivée/départ) a un impact sur la vitesse de l'équipe.

3

La vitesse baissera significativement avant de prendre une nouvelle valeur qui sera celle de la nouvelle équipe.

4

Le développement des compétences et de l'expertise des développeurs aura une influence positive sur la vitesse de l'équipe.

Tips

L'acquisition des compétences nécessaires et la maîtrise des valeurs métier permettront d'accroître la vitesse (+5%).



Rally Test Game



Objectif Qualité

Livrer une nouvelle version / release du Produit afin qu'elle soit stable et opérationnelle.

Rally Test Game

La conduite de projet agile requiert une activité de qualification des nouvelles fonctions par les représentants métier.

Réaliser un atelier RTG est une solution afin de répondre à la problématique des cas de tests non couverts par les tests de non-régression.

Conduite de l'atelier RTG

Prérequis : organiser et rédiger par rôle utilisateur l'ensemble des nouveaux cas de figures-métiers présents dans l'incrément fonctionnel à livrer.

Activité : les participants au RTG ont pour objectif de réaliser les missions qui leur sont confiées par rôle utilisateur

Durée : prévoir 1 à 3 jours en fonction de la taille de l'application

Livrable : bilan du RTG avec chaque anomalie fonctionnelle rédigée sous la forme d'un post-it dont la position ou la couleur dépendra de sa sévérité (critique, majeur, mineur).



Bateau agile

Speed Boat

Le Speed Boat est un « Innovation Game » : il fait partie des outils enseignables à tous.

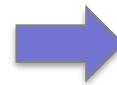


Le Speed Boat est une des pratiques agiles permettant de collecter du feedback sur un projet Agile.



L'amélioration continue d'un projet agile peut s'appuyer sur un Innovation Game Speed Boat.

Le Speed Boat peut aussi être utilisé afin d'animer la rétrospective d'un projet agile.



Faire un point de situation sur l'état d'un Produit ou sur l'état d'un Projet avec le Speed Boat fait partie des pratiques agiles.



Bateau agile

Speed Boat

Le Speed Boat est un « Innovation Game » : il fait partie des outils enseignables à tous.



Contexte



Le bateau Agile fait partie des « Innovation Games », de ces jeux qui nous rendent service dans les ateliers et les réunions afin de mener à bien les projets agiles.

Objectif



Le bateau Agile permet dans un format ludique, collectif et engageant, d'identifier les contraintes, les freins, les problèmes, de les prioriser afin d'identifier et de planifier les actions permettant de les résoudre.



Bateau agile

Speed Boat



Le bateau Agile, un bateau aux multiples facettes...
L'intérêt de cet Innovation Game est qu'il peut être utilisé dans plusieurs situations.

Le bateau peut représenter :

- **le projet**
- **le produit**
- **une équipe**

Evènement déclencheur

- Le projet n'avance pas pour telle ou telle raison.
- Le produit est soumis à des critiques et à des commentaires de la part des utilisateurs.
- L'équipe a besoin de soutien afin de promouvoir l'agilité dans l'organisation.



Favourable
winds

Positives leverages

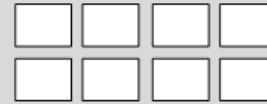


Vents
favorables

Les leviers positifs

Adverse
winds

Pitfalls to avoid



Vents
contraires

Les écueils à éviter

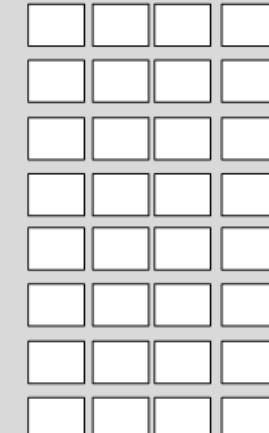
The Island
TOMORROW

Target
identification
on several axes

L'île

DEMAIN

identification de
la cible sur
plusieurs axes



The Dock

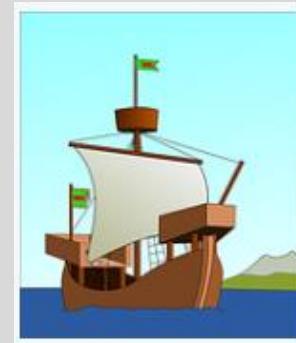
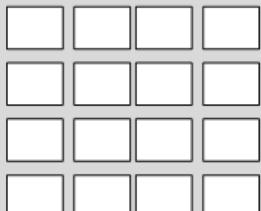
TODAY

The Basis

Le quai

AUJOURD'HUI

La base



Anchors



Obstacles &
bottlenecks

Ancres



Les freins



Bateau agile

Speed Boat

Le bateau symbolise le **produit** à réaliser.

- L'île est le symbole des objectifs et des résultats attendus du produit.
- Les vents favorables symbolisent les éléments fonctionnels et techniques qui sont déjà opérationnels et contribuent positivement à l'atteinte des objectifs du produit.
- Les vents défavorables symbolisent les contraintes fonctionnelles, techniques ou organisationnelles qui ne sont pas (encore) résolues.
- Les ancre symbolisent les freins qui nous ralentissent et nous empêchent d'atteindre notre île et donc les objectifs du produit.



1

Chacun des participants prend quelques minutes pour écrire sur des post-it les objectifs du produit à réaliser.

2

Chacun des participants vient à tour de rôle coller ses post-it au bon endroit sur le Speed Boat.

3

On poursuit avec les vents favorables ou défavorables et les freins qui ralentissent l'atteinte des objectifs à réaliser.

4

Enfin les participants votent pour les vents défavorables et les freins à résoudre en priorité.



Bateau agile

Speed Boat

Le bateau symbolise l'équipe Projet.

- L'île est le symbole des objectifs du projet à atteindre.
- Les vents favorables symbolisent les éléments qui ont aidés l'équipe à répondre à ses engagements : ce qui a facilité l'avancement du projet.
- Les vents défavorables symbolisent les contraintes fonctionnelles, techniques ou organisationnelles qui ne sont pas (encore) résolues.
- Les ancre symbolisent les freins qui empêchent d'atteindre l'île et donc de réaliser les objectifs du projet.



1

Chacun des participants prend quelques minutes pour écrire sur des post-it ses objectifs dans sa participation au projet.

2

Chacun des participants vient à tour de rôle coller ses post-it au bon endroit sur le Speed Boat.

3

On poursuit avec ce qui a été apprécié, ce qui a fait avancer le projet ou ce qui l'a freiné pour atteindre les objectifs des précédentes itérations.

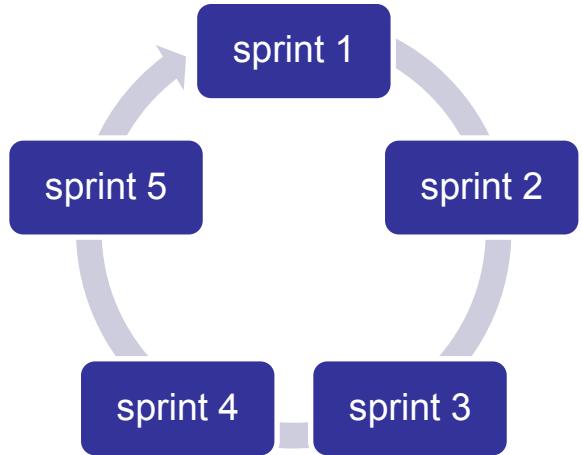
4

Enfin les participants votent pour les actions qu'ils souhaitent mettre en œuvre rapidement afin d'améliorer l'organisation et la performance du projet.



Méthodes agiles

Merci



Méthodes agiles

Méthode
SCRUM



Conduite de projet Agile
Equipe de réalisation



Lettre de mission

Development Team

Livrer un incrément de produit de qualité

Fonctionner de manière optimale avec 3 développeurs

Demander le soutien du Scrum Master en tant que de besoin

Etre force de proposition avec le Product Owner

L'équipe de réalisation est une équipe multidisciplinaire : elle est légitime pour s'organiser elle-même et gérer son travail avec le soutien du Scrum Master.

L'équipe de réalisation accueille toutes les compétences nécessaires afin de livrer un incrément de produit de qualité.

L'équipe de réalisation acquiert les bonnes pratiques et développe son expertise auprès du Scrum Master.

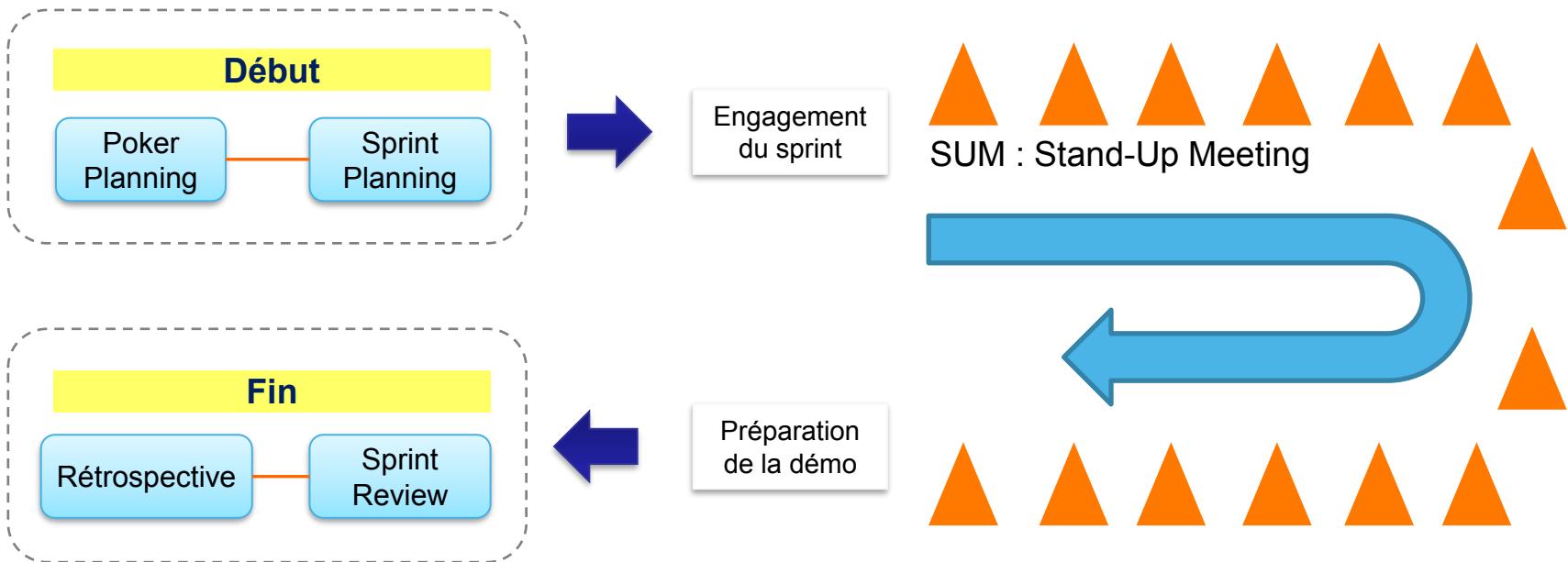
Les membres de l'équipe de réalisation sont appelés développeurs quel que soit le travail réalisé.

Les membres de l'équipe peuvent avoir des compétences spécialisées.

Il n'y a pas de sous-traitance ou de sous-équipe et cela quel que soit le domaine d'activité (design, testing, DevOps, ...).

La responsabilité globale demeure au niveau de l'équipe entière.

Accompagnement de l'équipe de réalisation



La vélocité de l'équipe de réalisation

- 1** La vélocité est une mesure de vitesse.

- 2** La vélocité indique la quantité de Product Backlog qui peut être traité à chaque itération.

- 3** La vélocité dépend du nombre de développeurs et de leur expérience.

- 4** La vélocité dépend de la durée des itérations.

Tips

La vélocité est utilisée par le Product Owner lors des exercices de planification.

Comment ça marche ?

- 1** Une tâche de référence est choisie par l'équipe de réalisation afin de représenter 1 point.
- 2** Les stories du projet seront évaluées à partir de cette tâche de référence lors de chaque Poker Planning.
- 3** La vitesse d'une équipe de réalisation est déterminée après les 2 ou 3 premières itérations du projet.
- 4** La vitesse de l'équipe de réalisation devient alors une constante qui correspondra à sa capacité à faire.

Tips

Chaque équipe agile fait l'apprentissage de sa propre vitesse.

Comment ça marche ?

1

Une modification par le projet agile de la durée de ses itérations a un impact sur la vitesse.

2

Un changement dans l'équipe (arrivée/départ) a un impact sur la vitesse de l'équipe.

3

La vitesse baissera significativement avant de prendre une nouvelle valeur qui sera celle de la nouvelle équipe.

4

Le développement des compétences et de l'expertise des développeurs exerce une influence positive qui permettra d'augmenter la vitesse de l'équipe.

Tips

L'acquisition des compétences nécessaires et la maîtrise des valeurs métier permettront d'accroître la vitesse (+5%).

Mesures de la performance

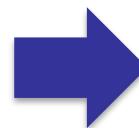


1

Mesure de la
performance du Done

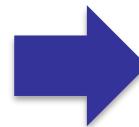
2

Mesure des coûts de
non qualité



SPRINT [1,N]

$$\frac{\Sigma (\text{nombre (stories Done)})}{\Sigma (\text{nombre (stories))})}$$



SPRINT [1,N]

$$\frac{\Sigma (\text{complexité (stories Anomalie)})}{\Sigma (\text{complexité (stories))})}$$

$$\frac{\Sigma (\text{complexité (stories))})}{\Sigma (\text{complexité (stories))})}$$

Comment ça marche ?

SPRINT [1,N]

1

Σ (nombre (stories Done))

 Σ (nombre (stories))



Il s'agit d'observer la capacité de l'équipe de réalisation à tenir sa promesse de livrer à la fin de chaque itération les stories évaluées lors du Poker Planning.

SPRINT [1,N]

2

Σ (complexité (stories Anomalie))

 Σ (complexité (stories))



Il s'agit d'observer la capacité de l'équipe de réalisation à tenir sa promesse de livrer régulièrement en production des incrément de qualité aux utilisateurs.

Contribution aux cérémonies agiles

DEV TEAM	L	C	A	N
POKER PLANNING		X		
SPRINT PLANNING				X
DAILY STAND-UP		X		
SPRINT REVIEW		X		
RETROSPECTIVE		X		

Légende

 Leader

 Auditeur

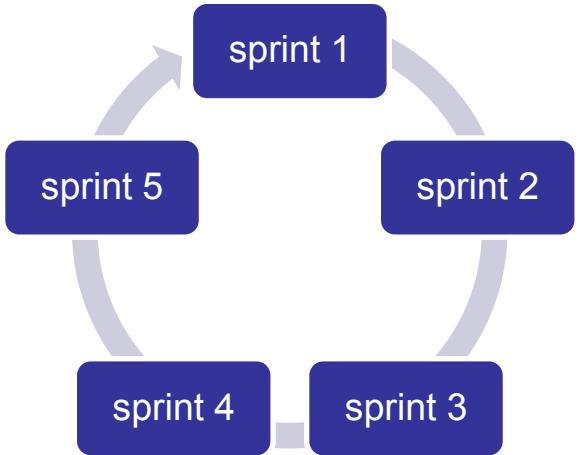
 Contributeur

 Ne participe pas

- Une nouvelle liste de stories sélectionnées par le Product Owner à partir du Product Backlog est présentée aux développeurs lors du Poker Planning.
- Les développeurs participent activement à l'évaluation des stories du prochain sprint.
- Les développeurs participent aux réunions quotidiennes (mêlées).
- Les développeurs s'engagent à livrer un incrément de produit de haute qualité.
- Les développeurs participent activement à la démonstration des stories du sprint.
- Les développeurs contribuent à l'amélioration continue du projet agile.

Les méthodes agiles

Merci



Méthodes agiles

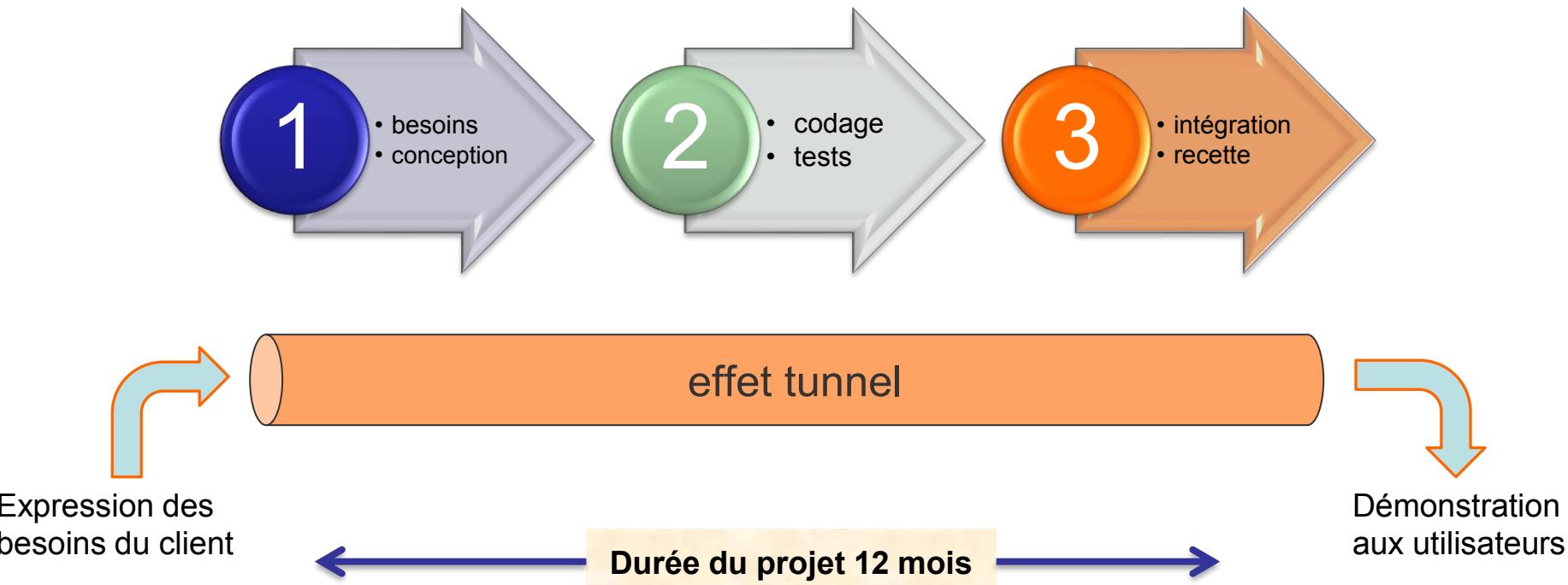
Méthode
SCRUM



Conduite de projet agile
Les utilisateurs



Effet tunnel des projets classiques



Effet tunnel des projets classiques

1

L'effet tunnel renvoie les utilisateurs sur une vision 12 mois en arrière.

2

Le marché, la technologie et les concurrents définissent les Produits de demain.

3

Les utilisateurs observent les écarts entre les produits.

4

Les utilisateurs privilégient les produits qui sont à leur écoute.

Les inconvénients des projets classiques

L'approche prédictive
n'aime pas les
changements
imprévus.

Pour les
développeurs, les
tests de validation
du client sont tardifs.

Pour les utilisateurs,
la recette est figée
sur l'expression
initiale des besoins.

Remise en cause de l'engagement
= Augmentation de la durée du projet
= Augmentation du budget

Pas de validation au fil de l'eau
= Pas de retour du client

Pas de feedback des utilisateurs
= Pas d'amélioration continue du Produit

Rappel - Le périmètre fonctionnel initial représente le socle d'un projet classique pour lequel le prestataire a une obligation de résultat.

Conduite de projet agile

1

livrer plus rapidement

2

livrer plus souvent

3

maximiser la satisfaction des utilisateurs

4

collecter le feedback des utilisateurs

Livrer plus rapidement



Observez les activités et les ressources engagées pour livrer en production **un nouvel incrément** aux utilisateurs.

- Finalisez des travaux de développement
- Exécutez des tests de non-régression
- Corrigez les anomalies observées
- Réalisez un RTG (Rally Test Game)



Rally Test Game

La conduite de projet agile requiert une activité de qualification qui nécessite la collaboration régulière ou "à la demande" des représentants métier.

Un atelier RTG est un moyen immédiat de répondre à la problématique projet des cas de tests non couverts par les tests de non-régression.

Conduite de l'atelier RTG

Prérequis : organiser par rôle utilisateur l'ensemble des nouveaux cas de figures-métiers présents dans l'incrément fonctionnel à livrer.

Durée : prévoir une à trois journées en fonction de la taille de l'application

Livrable : rédiger chaque anomalie fonctionnelle sous la forme d'un post-it dont la position ou la couleur dépendra de sa sévérité (critique, majeur, mineur).

Livrer plus souvent



Mesurez le temps et l'énergie nécessaires pour réaliser la livraison en production **d'un nouvel incrément.**



Satisfaction des utilisateurs

Identifiez la fréquence de livraison permettant de réaliser la collecte des retours des utilisateurs.

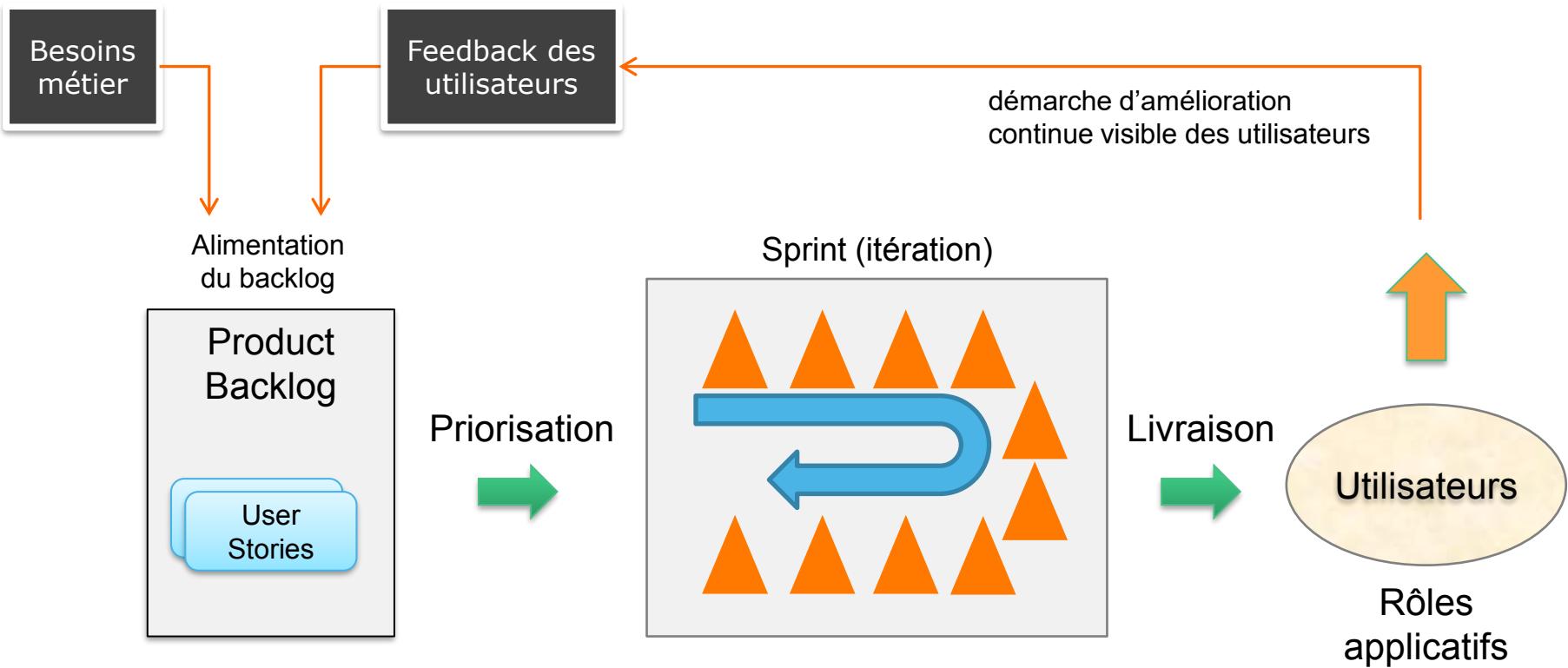
Vérifiez que la prise en compte du feedback des utilisateurs soit effective en termes de priorisation du Product Backlog afin que les utilisateurs puissent observer l'amélioration continue du produit.

Comparez le temps et les points nécessaires à une livraison par rapport à la durée d'une itération : **l'investissement** est-il satisfaisant au regard de la fréquence de livraison souhaité ?

Plan de livraison des versions / releases

L'effort consenti pour livrer doit également permettre de satisfaire les évènements et les dates clés retenues par le client afin d'accompagner la **Vision** du projet agile.

Collecter le feedback des utilisateurs



Pourquoi collecter le feedback des utilisateurs ?

1

co-construire le produit
avec les utilisateurs

2

répondre aux attentes
des utilisateurs

3

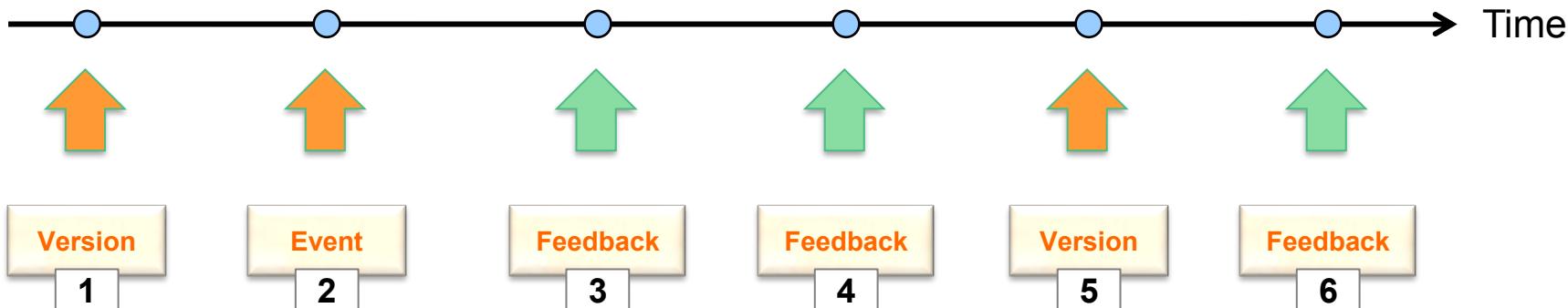
mettre en œuvre une
démarche
d'amélioration continue

4

maximiser la
satisfaction des
utilisateurs

Pourquoi livrer plus souvent ?

Livrailles à la demande



2 objectifs

a) Accomplissement de la Vision définie par le client

b) Démarche d'amélioration continue visible des utilisateurs

Comment livrer plus souvent ?

1

innover afin de savoir
livrer la valeur métier

2

organiser des tests de
non-régression

3

utiliser des services
Cloud (CaaS)

4

réduire le niveau
d'effort des livraisons

Utiliser l'approche CI/CD

Continuous Integration / Continuous Delivery

1 Exploration continue

2 Alimentation continue du backlog

3 Intégration continue

4 Livraison à la demande

L'exploration des besoins et des exigences est engagée et se poursuit pendant toute la durée du projet.

La rédaction des user stories est indispensable à la vie du projet agile.

L'intégration continue s'appuie sur DevOps et utilise des tests de non-régression automatiques.

Livrer plus souvent permet de traiter le feedback des utilisateurs et de maximiser la satisfaction des utilisateurs.

Product Owner

Product Owner

Développeurs

Utilisateurs

Les méthodes agiles

Merci

