



AGILE SCRUM FOUNDATION



Co-funded by
the European Union





Gestion de projet Agile

OBJECTIF DU MODULE

- Compréhension des Principes Fondamentaux de l'Agilité.
- Maîtrise des Fondamentaux de Scrum.
- Appliquer les pratiques Scrum lors de l'exécution d'un Sprint.
- Développer des compétences en gestion du travail en cours et en résolution d'obstacles.
- Maîtrise des Méthodes de Suivi et de Mesure Agile.

INTRODUCTION AUX CONCEPTS AGILE

agile vs Agile

Le mot « agile » est un adjectif signifiant : se déplacer rapidement et facilement, ou penser rapidement et de manière intelligente (dictionnaire Oxford Advanced Learner).

Cependant, « Agile » (avec un grand A) est utilisé pour désigner un groupe de cadres utilisés pour le développement et la gestion d'initiatives (programmes ou projets)

INTRODUCTION AUX CONCEPTS AGILES

Objectif principal

L'objectif principal des frameworks Agile est de rendre le développement plus rapide et plus fluide, et de créer un résultat plus satisfaisant pour le client.

PRINCIPES AGILES

01

Agilité dans les Projets

Les méthodologies agiles favorisent la flexibilité, l'adaptabilité et la collaboration dans la gestion de projet.

02

Valeurs Agiles

Les principes agiles mettent l'accent sur les individus et les interactions, la livraison de logiciels opérationnels, la collaboration avec les clients et la réponse au changement.

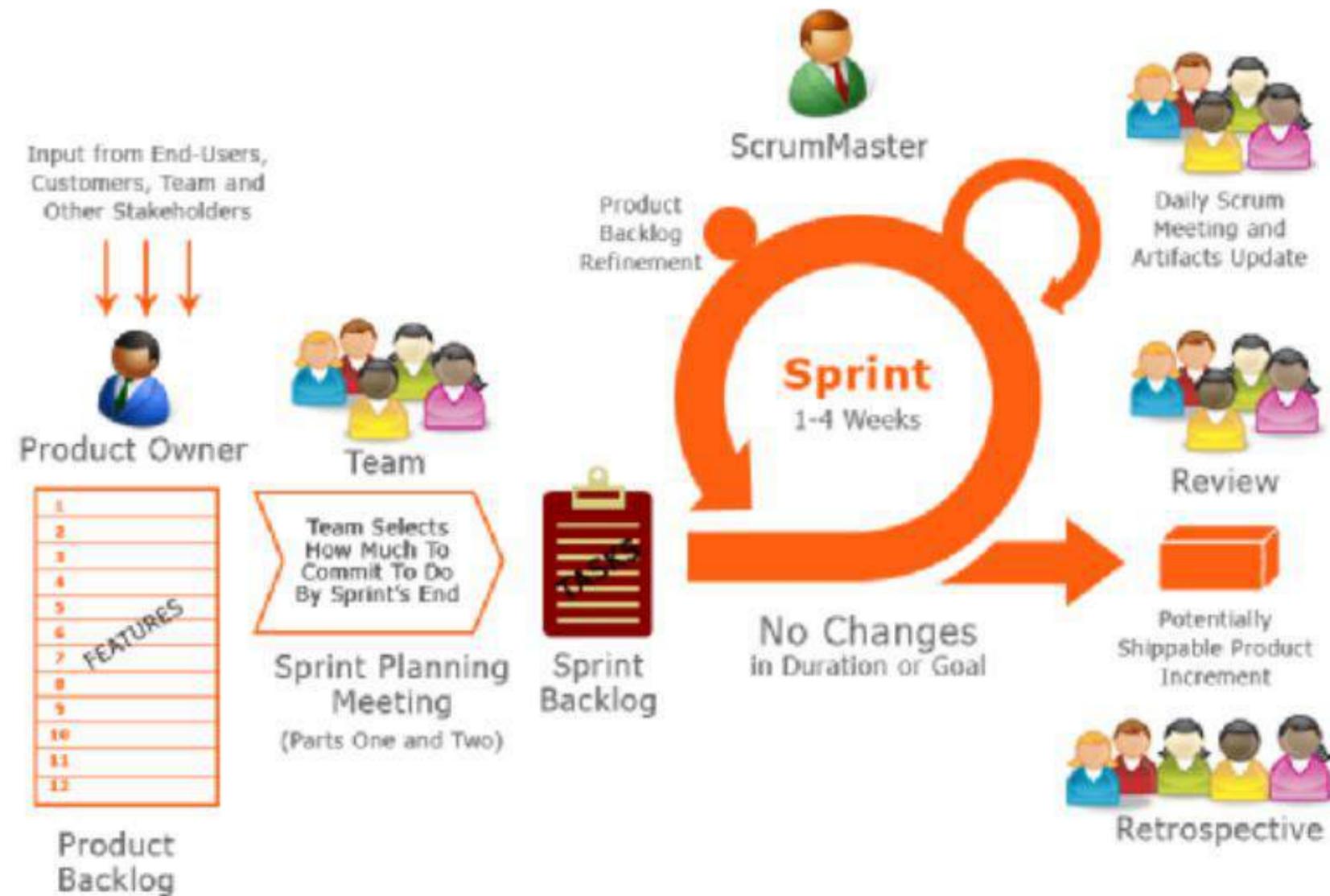
03

Cadres Agiles

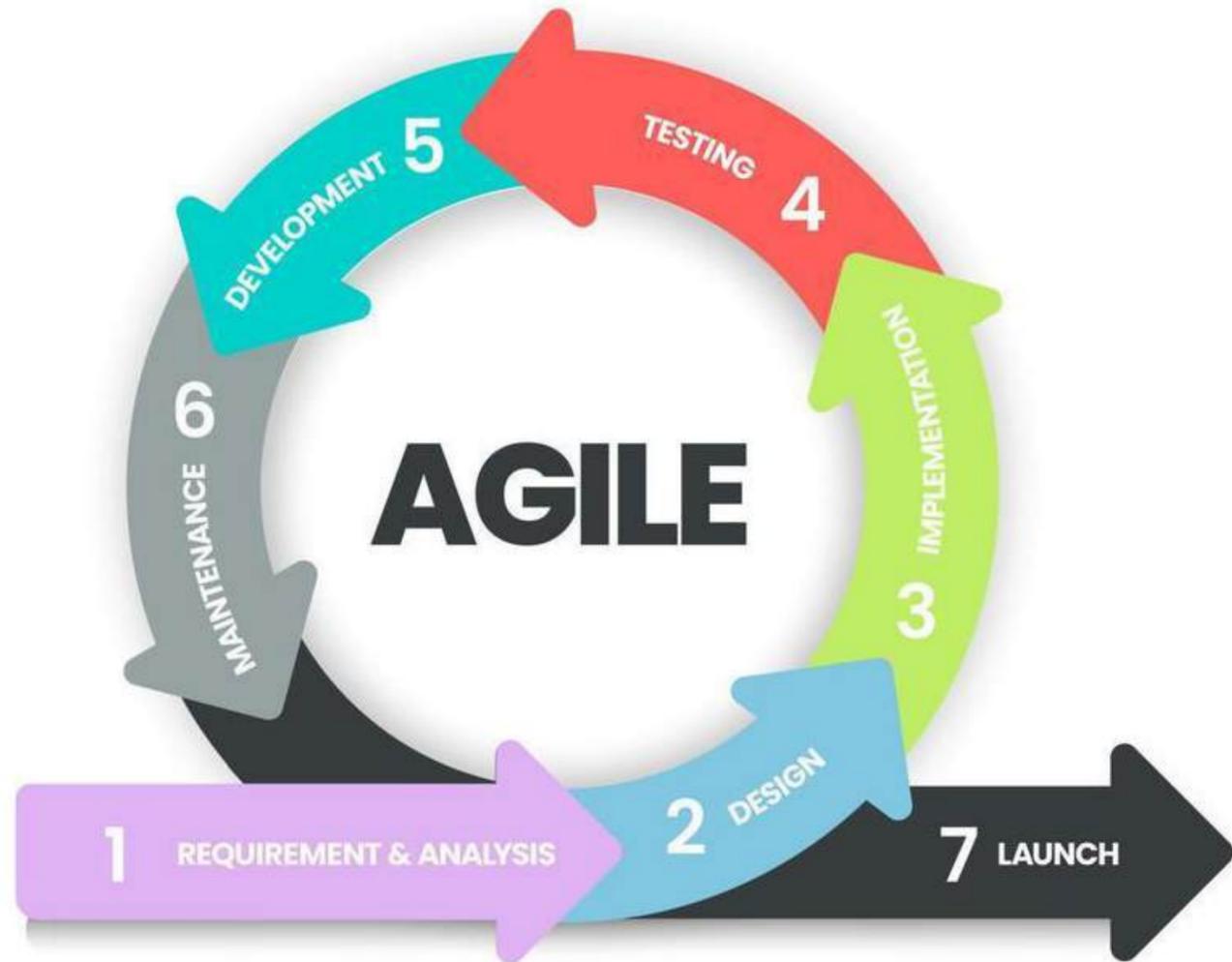
Différents cadres agiles tels que Scrum, Kanban et XP offrent des approches flexibles pour la gestion de projet.

PRINCIPES AGILES

Méthodes et cycle de vie du projet



Processus de développements courant pour un projet informatique :



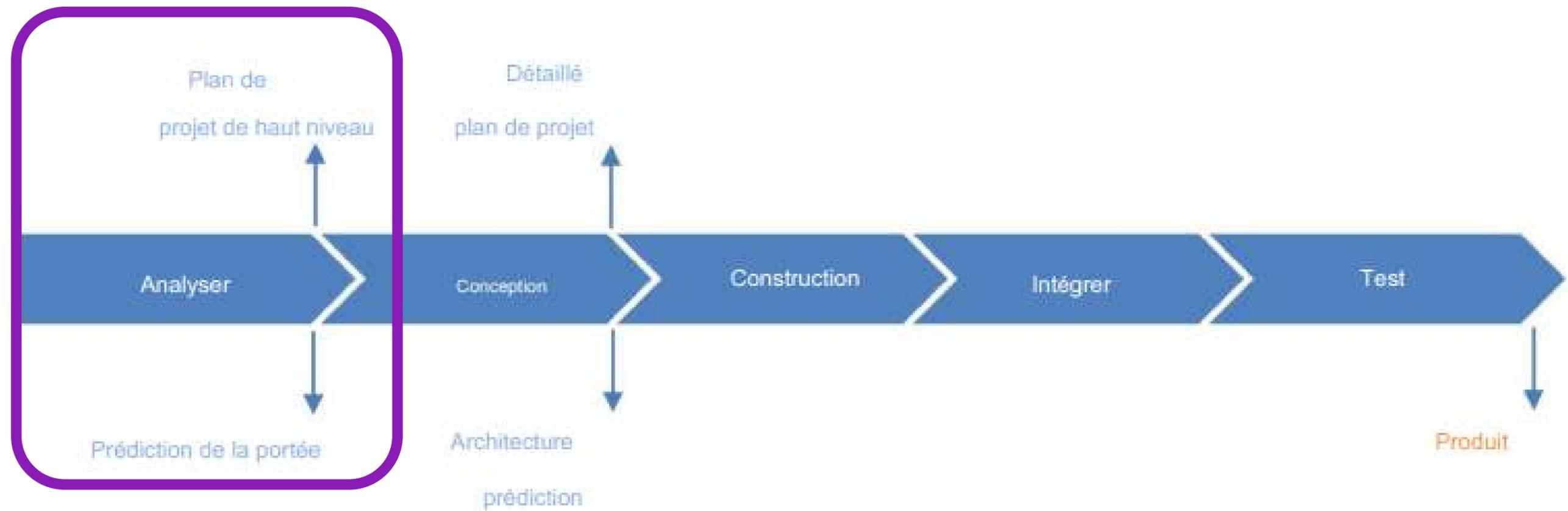
- Analyser
- Concevoir
- Construire
- Intégrer
- Tester

Ces processus sont requis pour chaque projet informatique : agile ou traditionnel

MÉTHODES ET CYCLE DE VIE DU PROJET

Il y a différentes manières dont nous pouvons concevoir le cycle de vie de développement sur la base des processus de développement standard.

Une façon d'exécuter les processus consiste à les exécuter les uns après les autres, comme le montre la figure suivante :

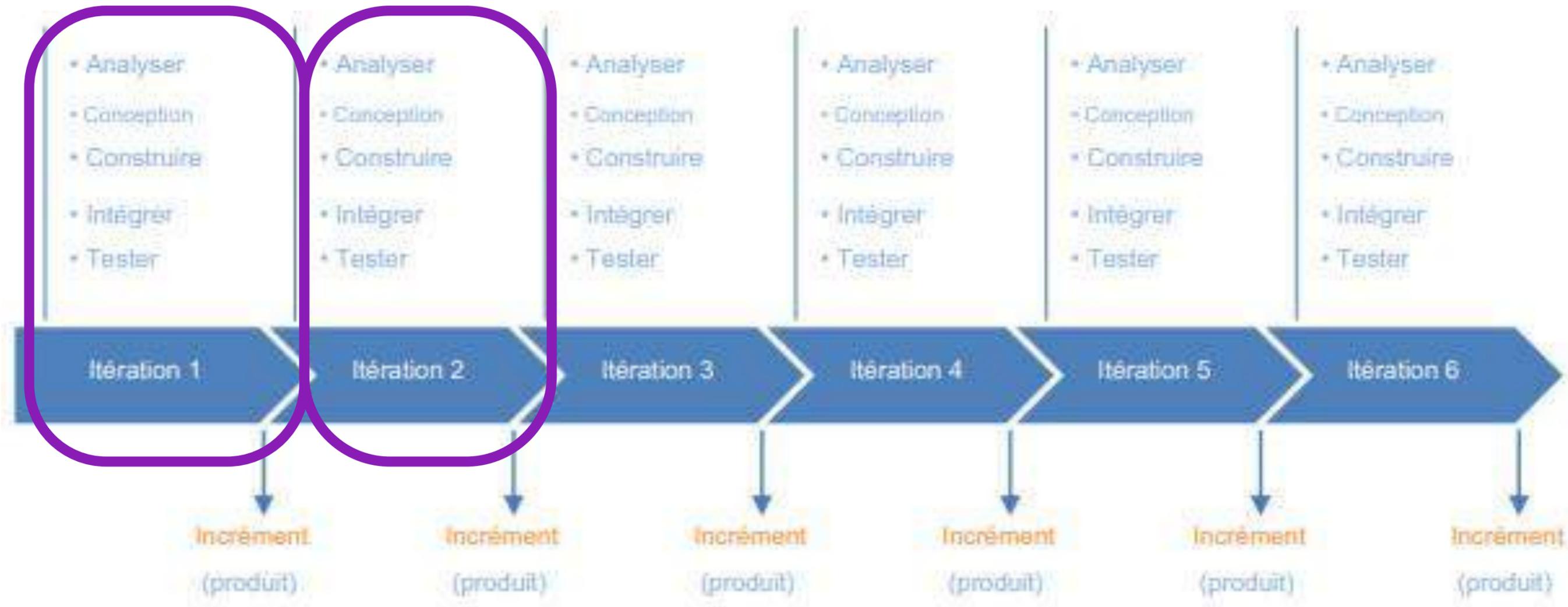


1. Analyser : spécifier les exigences, document appelé “requirements”
2. Design/Conception : concevoir l’architecture de la solution
3. Construction : construire le “logiciel” dans son intégralité
4. Intégrer : intégrer le code
5. Test : tester et déboguer lors de la dernière étape

MÉTHODE TRADITIONNELLE “WATERFALL”

- Tout est défini avant le début du développement.
- Cycle de vie en cascade
- Méthode prédictive
- Cycle de vie prédictif
- Projet exécuté étape par étape
- Idéal pour construction d'une maison
pas pour projet informatique
- Plan VS besoins du client





Plusieurs itérations appelés sprints dans Scrum

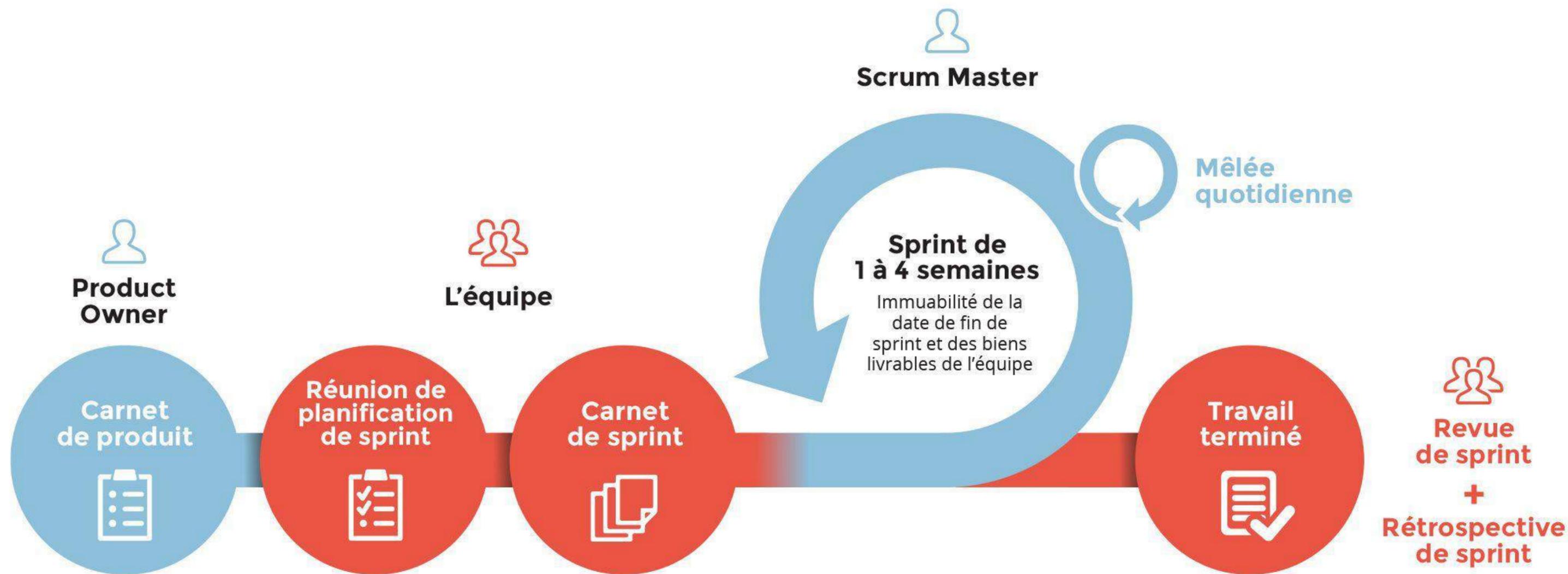
Tout le processus (1-4) sur une partie du produit à chaque itération

MÉTHODE ADAPTATIF “SCRUM”

- Afin d’avoir des solutions provisoires à la fin de chaque itération, de les montrer au client, de recevoir ses commentaires, d’appliquer ces remarques aux itérations suivantes...
- Solutions provisoires (incrément)
- Cycle de vie adaptatif
- Agile = utilisation de cycles de vie adaptatifs



MANIFESTE AGILE



MANIFESTE

Individus et interactions sur

Logiciel fonctionnel sur

Collaboration client sur

Répondre au changement sur

Processus et outils

Documentation complète

Négociation de contrat

Suivre un plan

MANIFESTE AGILE

12 principes accompagnent le Manifeste Agile :

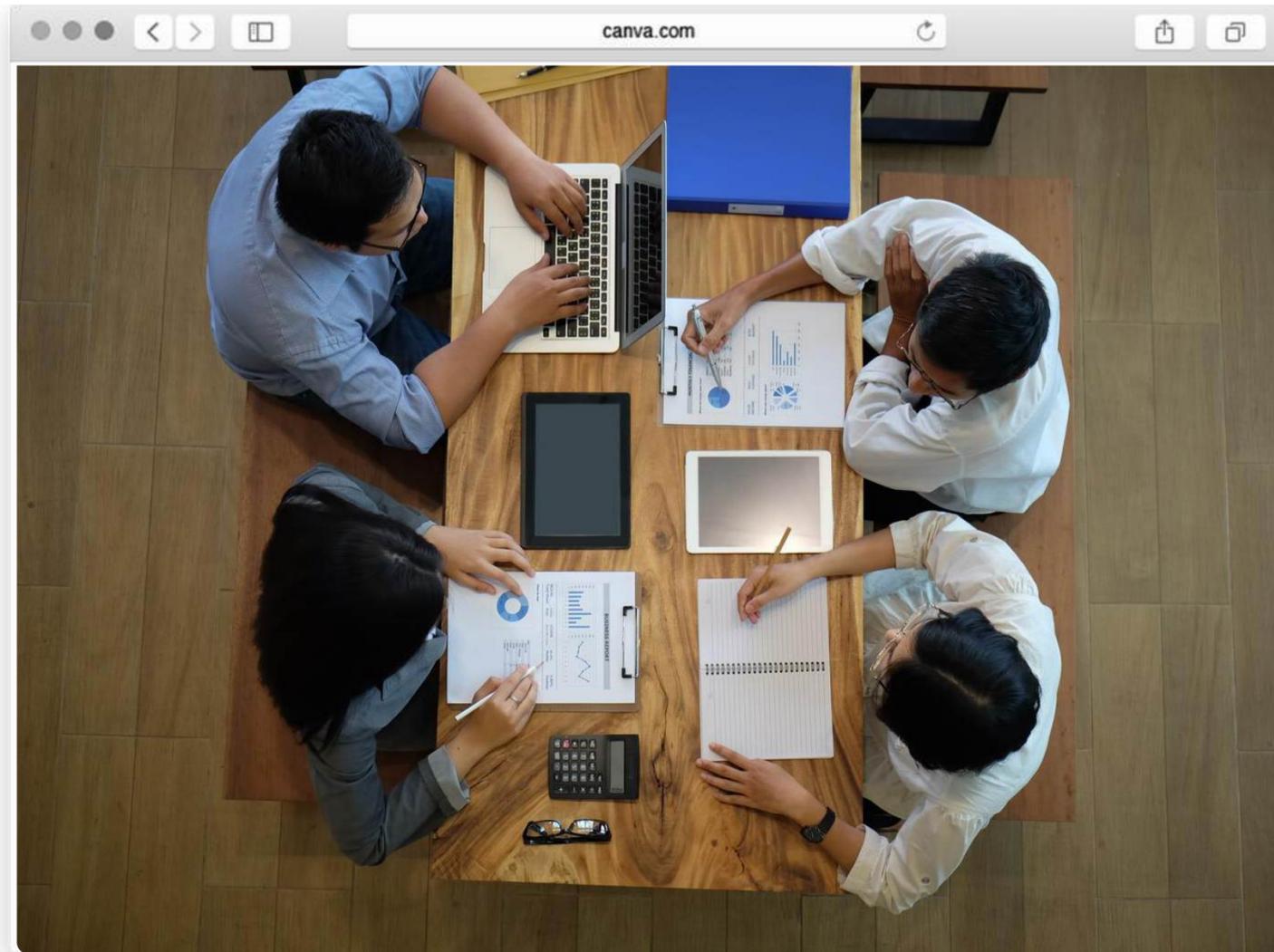
1. Notre priorité absolue est de **satisfaire le client** de manière précoce et continue.
2. Acceptez **l'évolution des exigences**, même tard dans le développement.
- 3. Livrer fréquemment** des logiciels fonctionnels.
4. Les hommes d'affaires et les développeurs doivent travailler **ensemble quotidiennement** tout au long du projet.
5. Construire des projets autour d'individus motivés. Donnez-leur l'environnement et le soutien dont ils ont besoin, et faites-leur **confiance pour faire le travail**.
6. La méthode la plus efficace et la plus efficace pour transmettre des informations vers et au sein d'une équipe de développement est la **conversation en face à face**.

MANIFESTE AGILE

7. **Un logiciel fonctionnel** est la principale mesure du progrès.
8. Les processus agiles favorisent le développement durable. Les sponsors, développeurs et les utilisateurs devraient pouvoir maintenir un **rythme constant** indéfiniment.
9. Une **attention continue** à l'excellence technique et à une bonne conception améliore l'agilité.
10. La simplicité – l'art de maximiser la quantité de travail non effectué – est essentielle. Les meilleures architectures, exigences et conceptions émergent de **l'auto-organisation des équipes**.
11. À intervalles réguliers, l'équipe réfléchit aux moyens de **devenir plus efficace**, puis ajuste et ajuste son comportement en conséquence.

Gérer le “scope”

Ce concept est généralement appelé Product Backlog dans les projets Agile. Un Product Backlog contient un certain nombre d’« éléments », et chaque élément décrit un élément constitutif de la solution finale.



PRODUCT BACKLOG

3 alternatives pour définir les éléments du Product Backlog :

1. **Spécification des exigences** (Requirement specification) :

- Manière la plus traditionnelle de définir la portée.
- Constitué de termes techniques clairs pour l'équipe de développement
- Basé sur les relations architecturales entre les éléments.

2. **Cas d'utilisation** (Use case) :

- Basé sur la compréhension et les besoins de l'utilisateur plutôt que sur une description technique
- Décrit un scénario détaillé avec toutes les actions qu'un utilisateur imaginaire fera et les comportements attendus de la solution.

PRODUCT BACKLOG

3 alternatives pour définir les éléments du Product Backlog :

3. **User story** :

- Similaire à un cas d'utilisation mais ne contient pas tous les détails
- Se concentre uniquement sur ses besoins et son objectif

Exemple : en tant qu'utilisateur final, je souhaite recevoir une liste de mes dernières transactions. Ou, en tant qu'administrateur système, je souhaite réinitialiser les mots de passe.

En tant que....., Je veux....., Afin de.....

2 règles concernant les éléments du Product Backlog

Compréhensibles



Ils ne doivent pas être techniques. En effet, le Product Backlog est notre outil de communication avec le client non technique. Normalement, un enfant de dix ans devrait être capable de comprendre chaque user story.

Indépendants



Ils doivent être indépendants les uns des autres, afin que nous puissions les développer librement. Aucun ordre particulier.

PRODUCT BACKLOG

- Fonctionnalités fonctionnelles et non fonctionnelles
- Bugs
- Spikes (pic)
- Priorisation MoSCow :
 - **M (Must Have)** > fonctionnalité qui DOIT figurer dans le produit final
 - **S (Should Have)** > caractéristique importante pour le produit final
 - **C (Could Have)** > fonctionnalité utile que nous aimerions avoir
 - **W (Will not Have This Time)** > fonctionnalité intéressante mais n'allons pas y investir maintenant

PRIORISATION MOSCOW

Méthode de priorisation MoSCoW

Mo

Doit être

Caractéristiques

Caractéristiques

Caractéristiques

S

Devrait être

Caractéristiques

Caractéristiques

Co

Pourrait être

Caractéristiques

Caractéristiques

Caractéristiques

Caractéristiques

Caractéristiques

W

Ne sera pas

Caractéristiques

PRIORISATION MOSCOW - EXEMPLE LA VOITURE

- Priorisation MoSCow :
 - **M (Must Have)** > Les freins
 - **S (Should Have)** > La climatisation
 - **C (Could Have)** > Une caméra et un écran pour la marche arrière
 - **W (Will not Have This Time)** > La surveillance en ligne de l'état mécanique de la voiture

Excellent moyen de se concentrer sur la valeur commerciale
>> se concentrer sur les besoins réels plutôt que sur les fonctionnalités sophistiquées (éléments qui pourraient avoir).

PRODUCT BACKLOG

- Principe du “Time box” utilisé dans les frameworks Agile
 - 2 caractéristiques :
 - Une durée maximale prédéfinie
 - Un ensemble prédéfini et généralement figé d’éléments

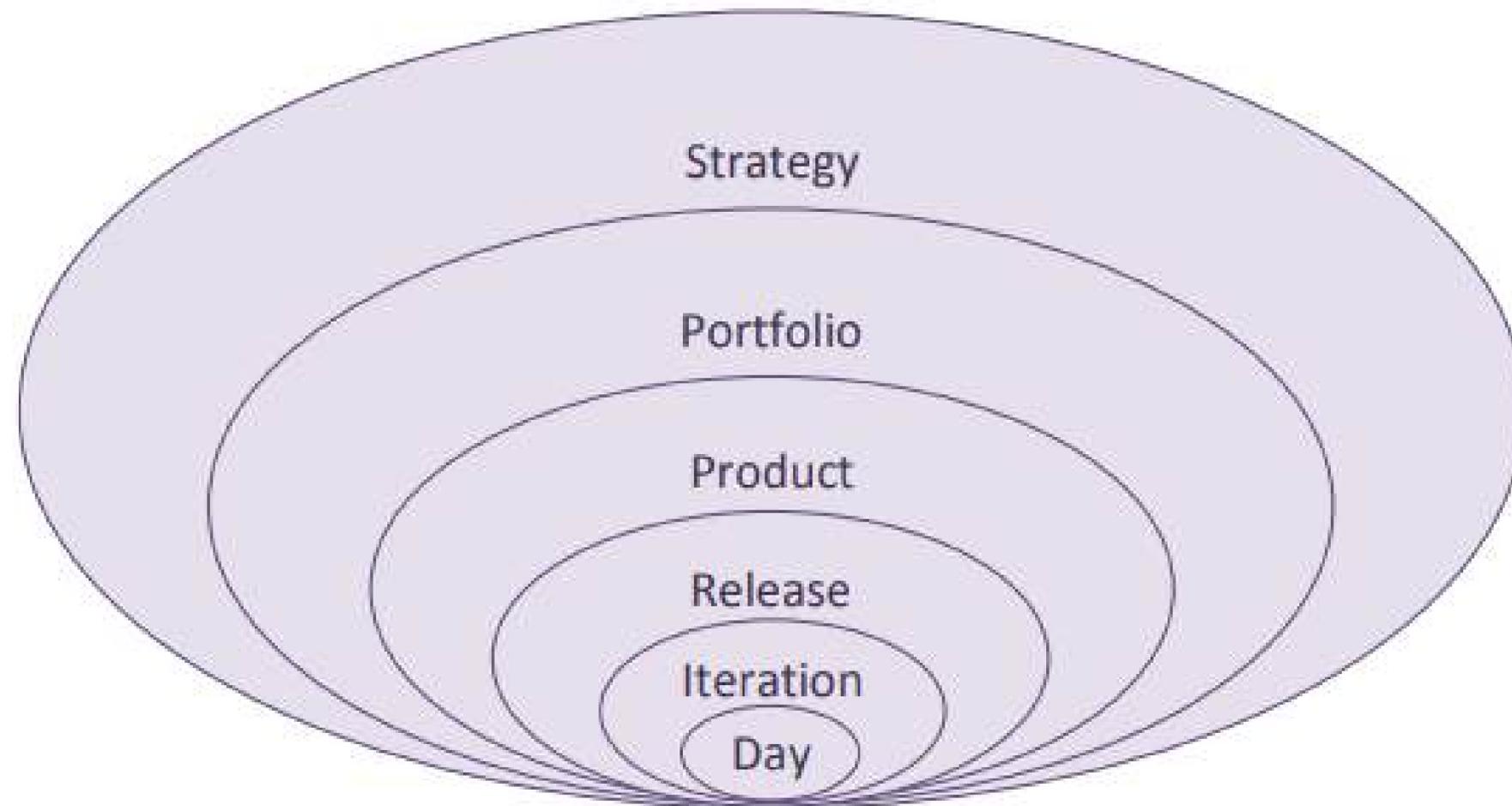
Chaque itération est limitée dans le temps :

- Délais de durée fixe
- Réunions internes + externes aux itérations (sprint planning, ...)

Augmente la productivité

PRODUCT BACKLOG

- Planning Onion : différents niveaux de planification liés à un projet Agile



4 NIVEAUX CONCEPTUELS DE PLANIFICATION

1. Planification du produit (product planning/
backlog product)
2. Planification des versions (release planning)
3. Planifications des itérations (sprint planning)
4. Planification quotidienne (daily scrum)

PRATIQUES AGILES

- 1. Programmation en binome** : conducteur & navigateur
- 2. Facteur de bus**
- 3. Développement piloté par les tests** : master test plan
- 4. Intégration continue**
- 5. Refactorisation continue** (amélioration du code)
- 6. Propriété collective du code** (tout le monde est responsable)
- 7. Rythme constant et durable**



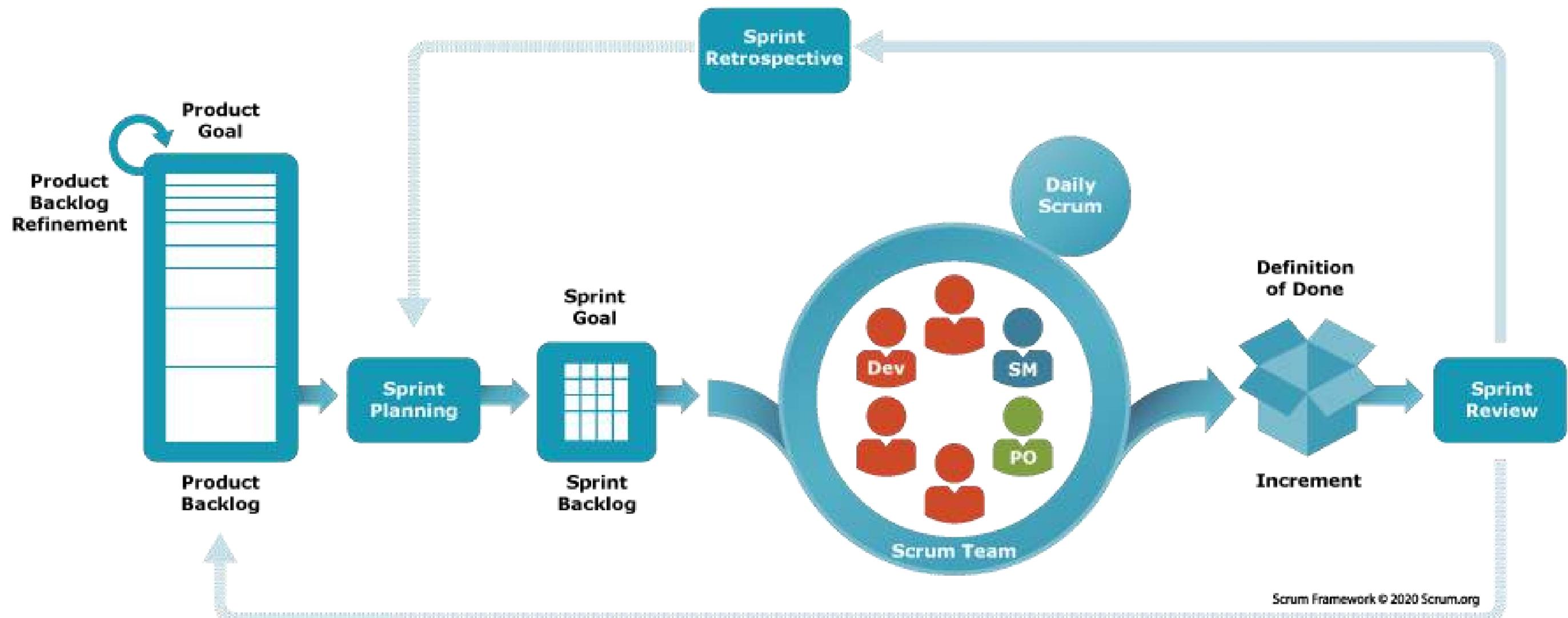
Questions ?



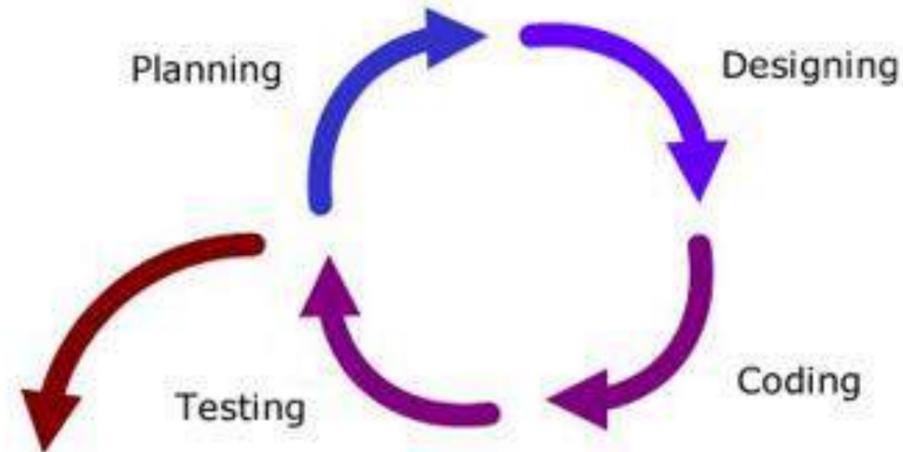
Cadre Scrum

CADRE SCRUM - COMMENT FAIRE FONCTIONNER L'AGILITÉ

SCRUM FRAMEWORK

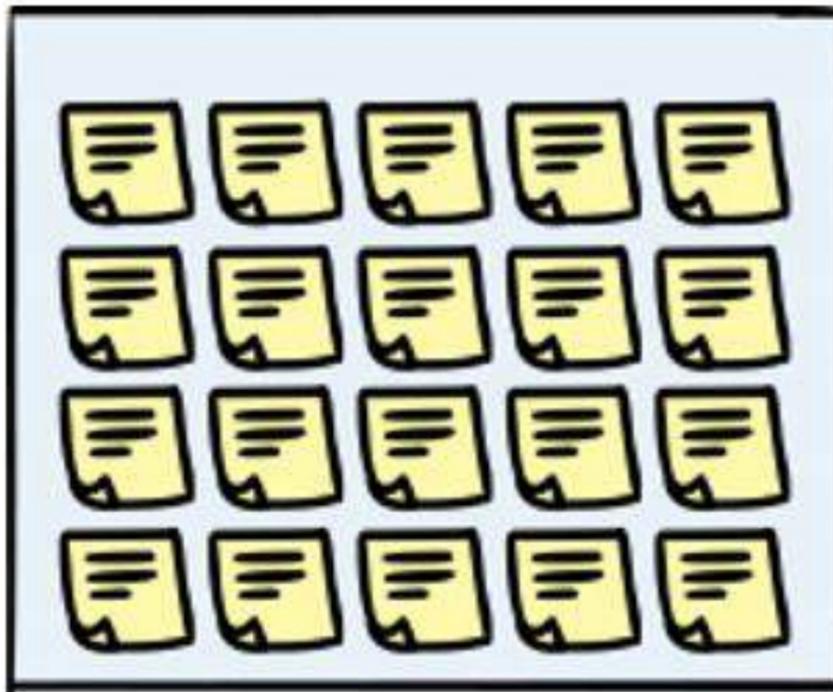


PLANIFICATION DE SPRINTS



Objectifs de Sprint

La planification de sprint définit les objectifs à court terme et les livrables attendus.



Backlog de Sprint

Les éléments du backlog de sprint sont sélectionnés pour être livrés dans le sprint.

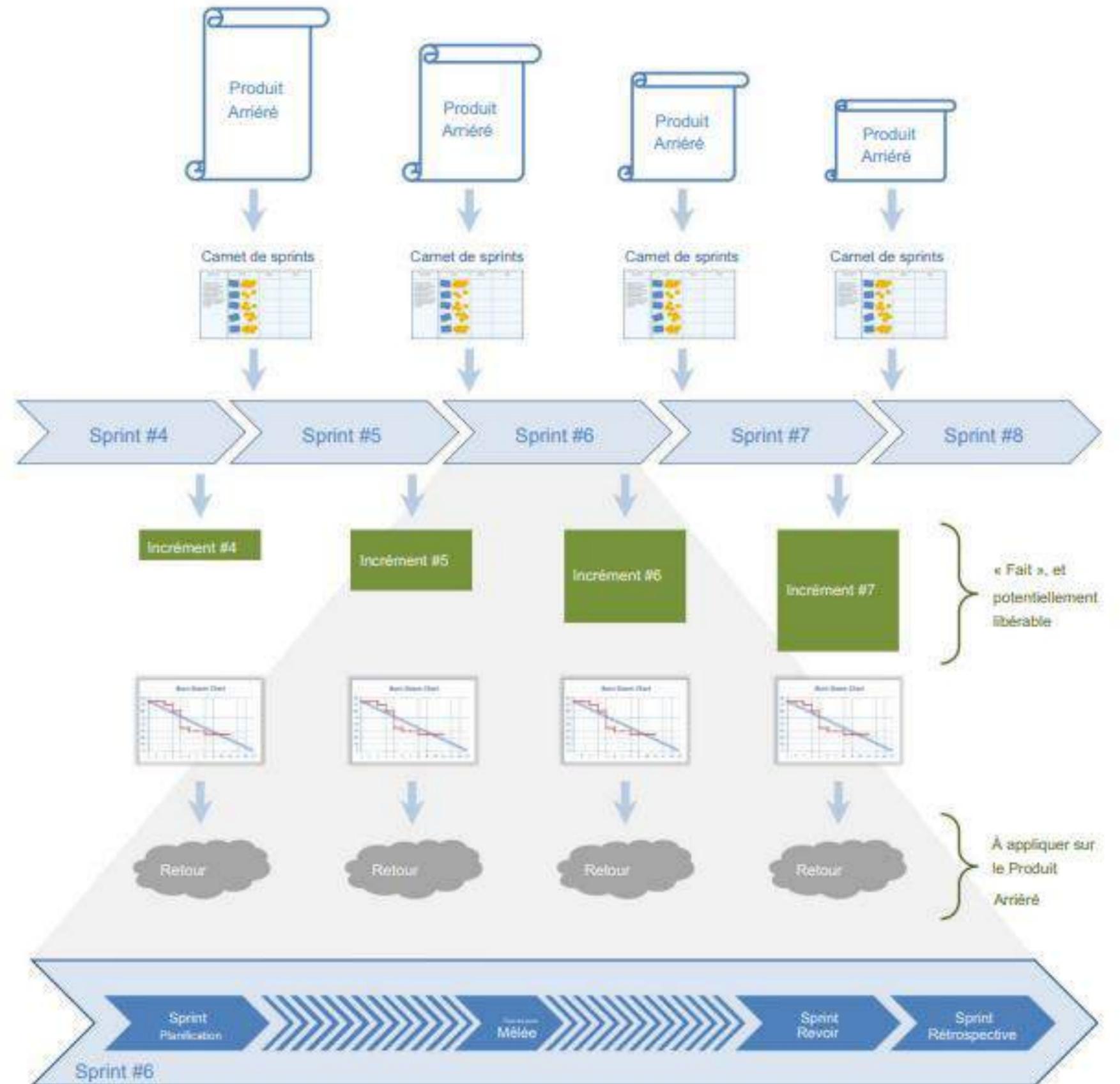


Engagement de l'Équipe

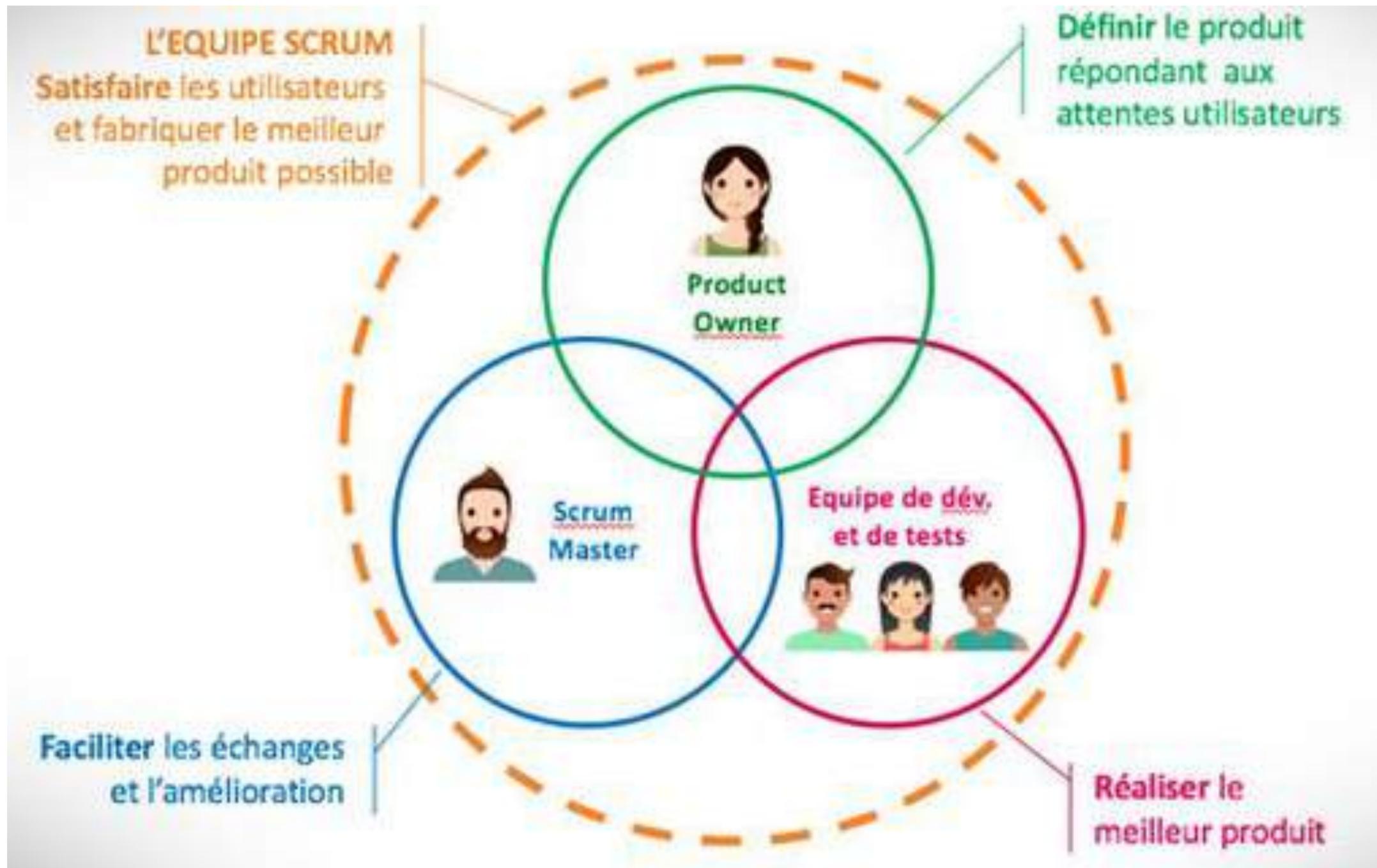
L'engagement de l'équipe est crucial pour la réussite du sprint.

CADRE SCRUM

- Sprint Planning
- Daily Scrum
- Sprint Review
- Rétrospective du sprint



3 ROLES DANS SCRUM



EQUIPE SCRUM - CARACTERISTIQUES

- **Auto-organisée**
- Est **pluridisciplinaire** et ne comporte pas de rôles prédéterminés (transversal)
- Est **responsable** de son engagement
- Possède l'autorité nécessaire pour **agir de manière à respecter ses engagements**
- Travaille dans des locaux ouverts et avoisinants (**communication**)
- **Résout** ses propres conflits
- Observe des règles de base de **fonctionnement et de comportement**



QUI EST CHEF DE PROJET ?

Ce rôle n'existe pas !
dans Scrum

Les responsabilités de gestion de projet sont réparties entre les 3 rôles de Scrum et il n'y a pas de gestion de projet centralisée dans Scrum

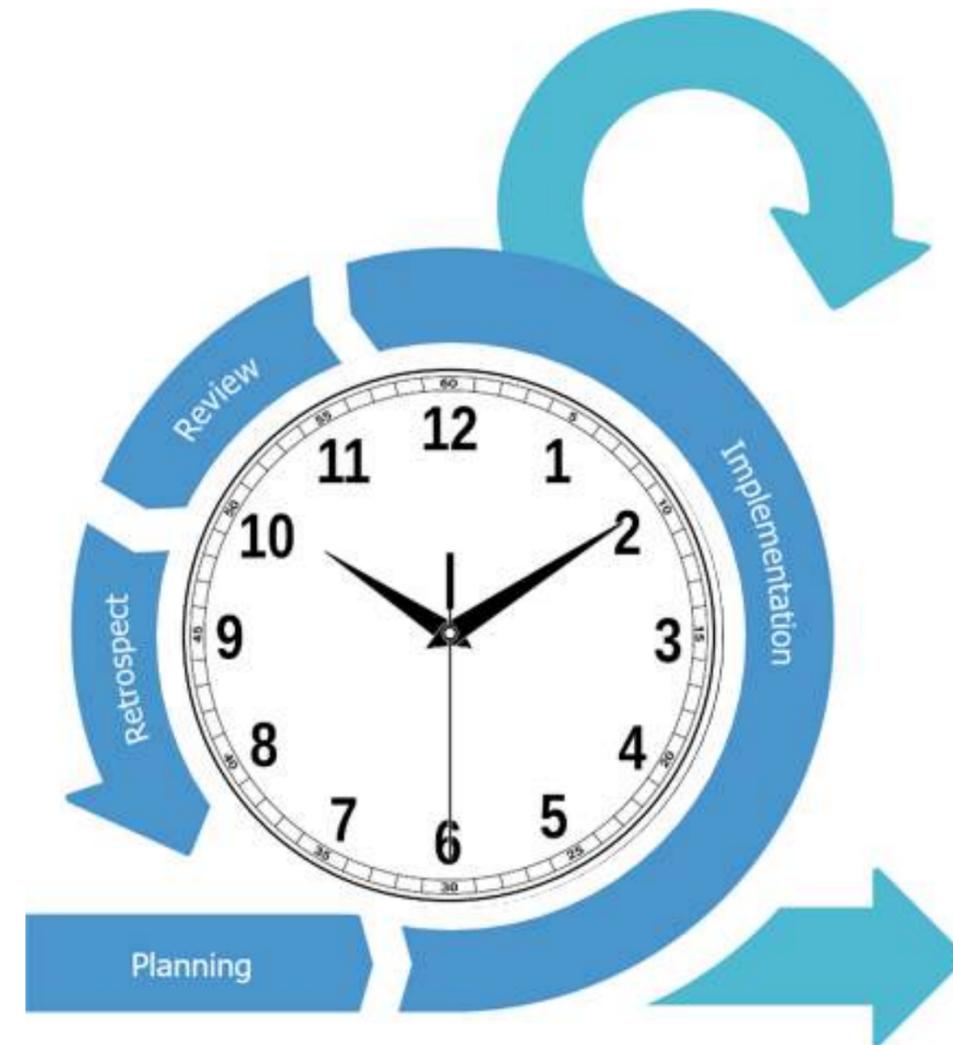
LES EVENEMENTS DANS SCRUM



METHODOLOGIE SCRUM - TIMEBOXING

TIMEBOXES FOR THE 5 SCRUM EVENTS

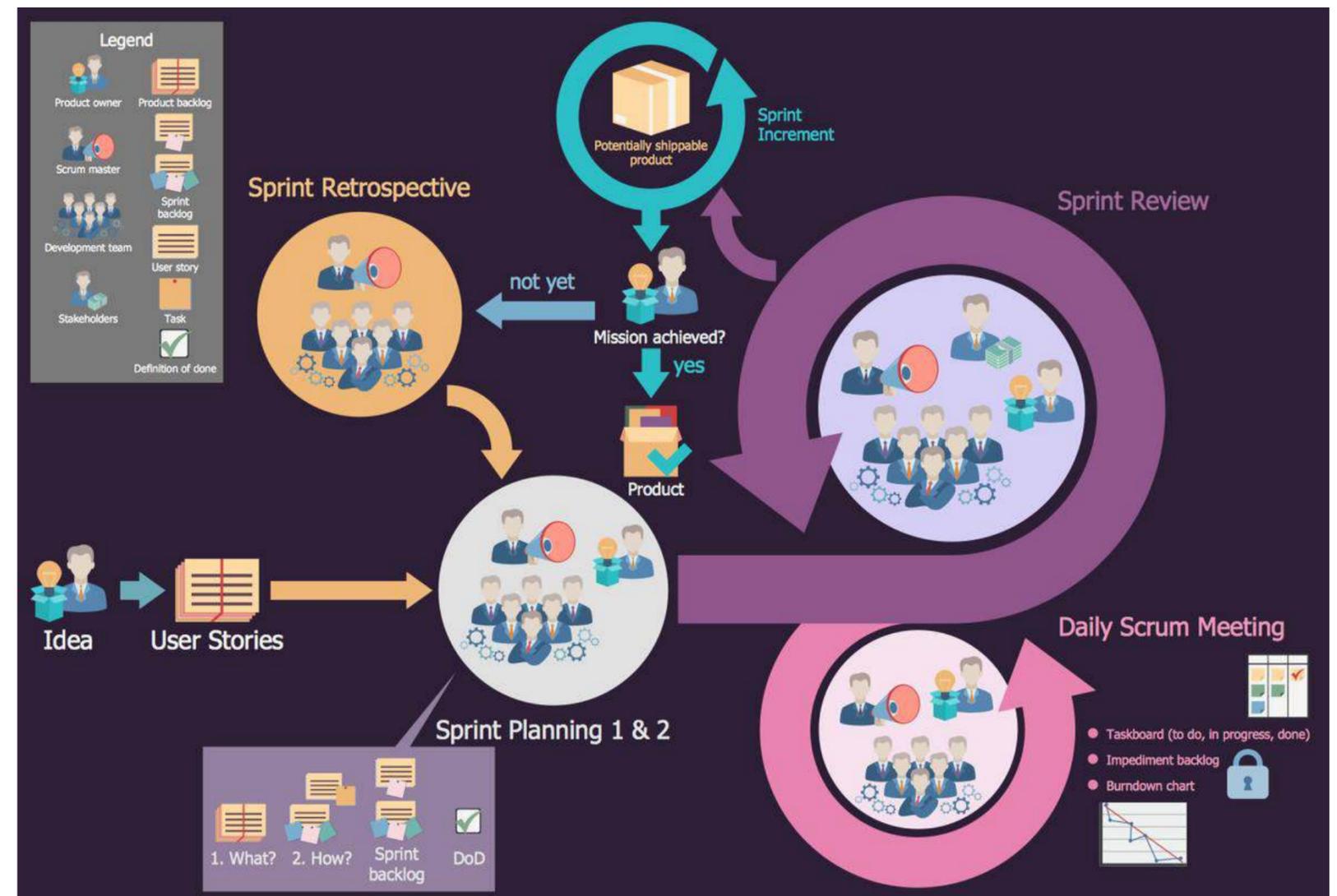
EVENT:	TIMEBOX:	FREQUENCY:
SPRINT	1 MONTH OR LESS	N/A
SPRINT PLANNING	EIGHT HOURS. FOR SHORTER SPRINTS, THE EVENT IS USUALLY SHORTER.	ONCE AT THE BEGINNING OF THE SPRINT
DAILY SCRUM	15 MINUTES	DAILY
SPRINT REVIEW	4 HOURS. FOR SHORTER SPRINTS, THE EVENT IS USUALLY SHORTER.	ONCE AT THE END OF THE SPRINT
SPRINT RETROSPECTIVE	3 HOURS. FOR SHORTER SPRINTS, THE EVENT IS USUALLY SHORTER.	ONCE AT THE END OF THE SPRINT



METHODOLOGIE SCRUM - WORKSPACE

L'équipe doit être regroupée dans une seule pièce.

Cela améliore les relations entre les membres de l'équipe et facilite leur collaboration



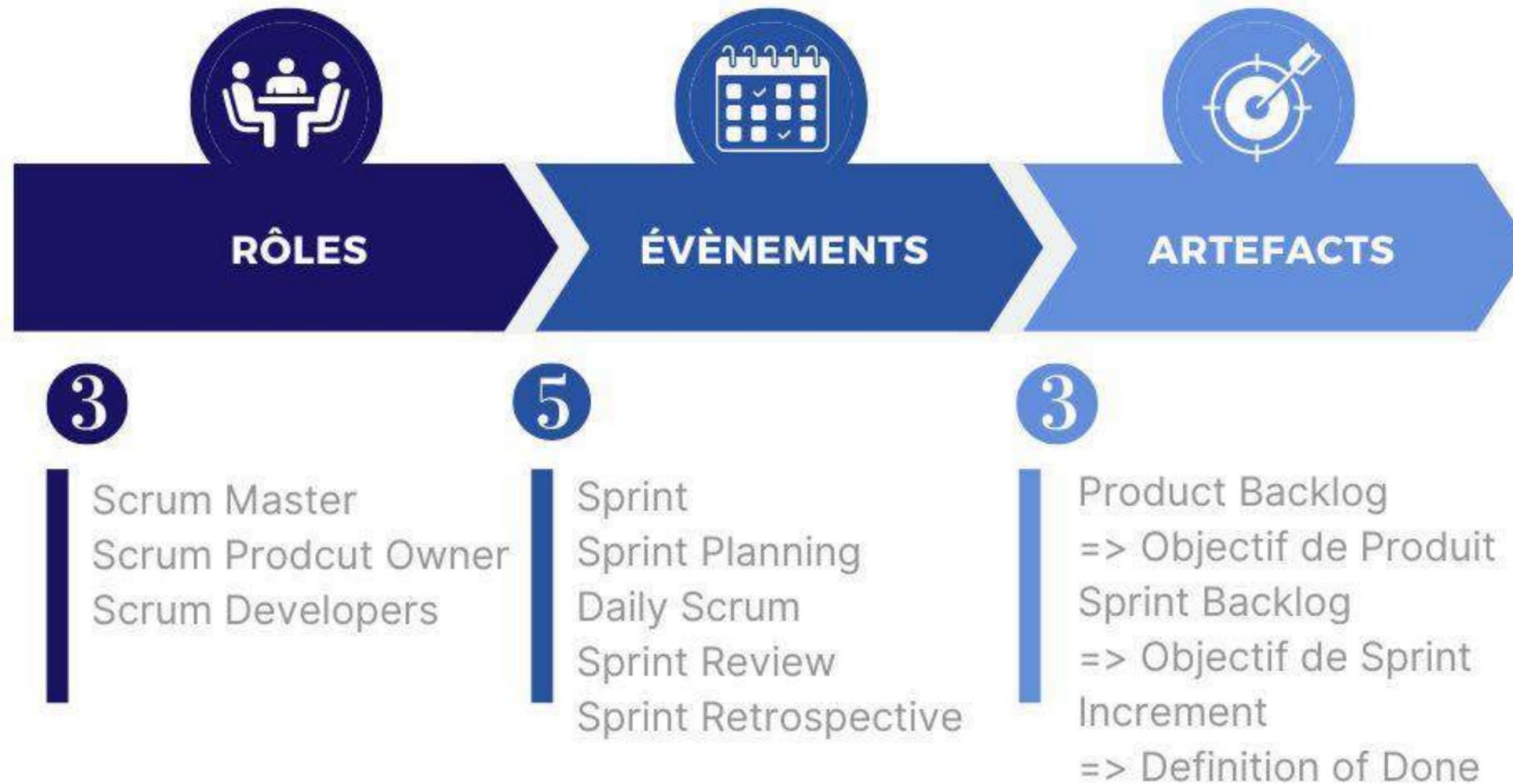
METHODOLOGIE SCRUM

- **Communication osmotique** : exemple chaque fois que vous envoyez un email, mettez tout le monde en copie.
- **Gestion d'équipe distribuées** : localisée au meme endroit sinon utiliser les outils digitaux.

LES ARTEFACTS SCRUM



LES ARTEFACTS SCRUM



ARTEFACT 1 : PRODUCT BACKLOG

- Product Backlog
- Elements du product backlog :
 - User stories
- Estimation :
 - Story points
 - Planning Poker
 - Triangulation
 - Estimation d'affinité
 - Ré-estimation

ARTEFACT 2 : SPRINT BACKLOG

- Éléments sélectionnés en haut du Product Backlog
- Objectif de Sprint
- Sprint planning
- Rapidité : vélocité

Objectif de sprint	Faire	Faire	Fait	
<p>Le but de ce sprint est de rendre la partie achats du site suffisamment mature pour pouvoir gérer les</p> <p>l'ensemble du processus et les utilisateurs peuvent expérimenter un processus d'achat complet, à travers lequel d'autres fonctionnalités du site internet sera plus significatif.</p> <p>Graphique de déroulement du sprint</p>	Item #1		Item #1 t.1.6 t.1.1 t.1.3 t.1.5 t.1.2	
	Item #2	t.2.7	t.2.6 t.2.5	t.2.1 t.2.3 t.2.2 t.2.4
	Item #3	t.3.4 t.3.5 t.3.3	t.3.2	t.3.1
	Item #4	t.4.4 t.4.2 t.4.5		
	Item #5			

Sprint 1	Sprint 2	Sprint 3	Sprint 4 105	Sprint 5	Sprint 6 110	Sprint 7
80sp	70sp	95sp	vitesses	130sp	vitesses	120sp

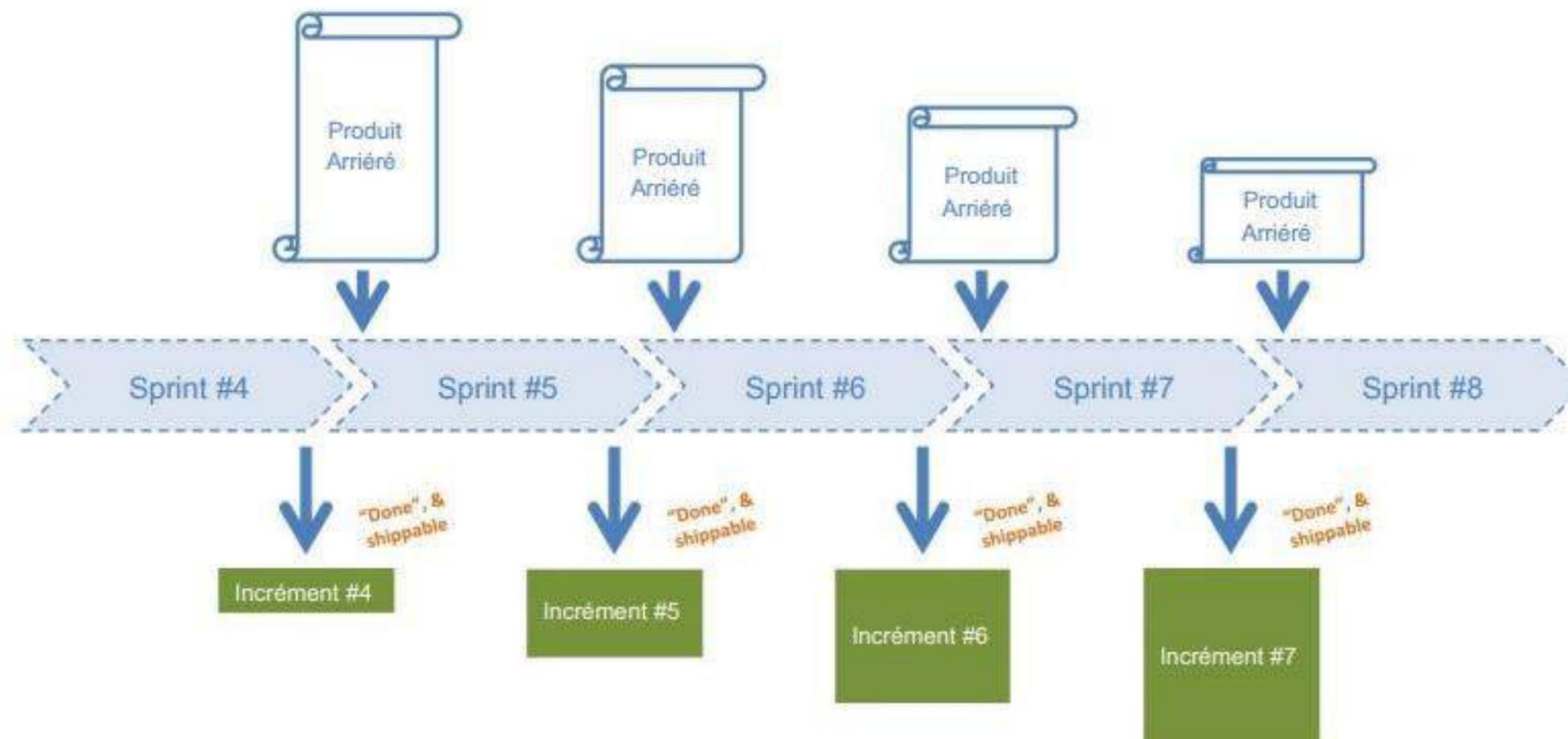
ARTEFACT 2 : SPRINT BACKLOG

- Rapidité : vélocité

Sprint 1	Sprint 2	Sprint 3	Sprint 4 105	Sprint 5	Sprint 6 110	Sprint 7
80sp	70sp	95sp	vitesses	130sp	vitesses	120sp

ARTEFACT 3 : INCRÉMENT

- Somme de tous les éléments du Product Backlog terminés à la fin d'un Sprint.



ARTEFACT 4 : DÉFINITION DE “TERMINÉ”

- DoD = Definition of Done
- Processus de développement
- Processus organisationnel
- Exigences non fonctionnelles
- Critères de qualité
- Documentation Agile
- Tests Agiles

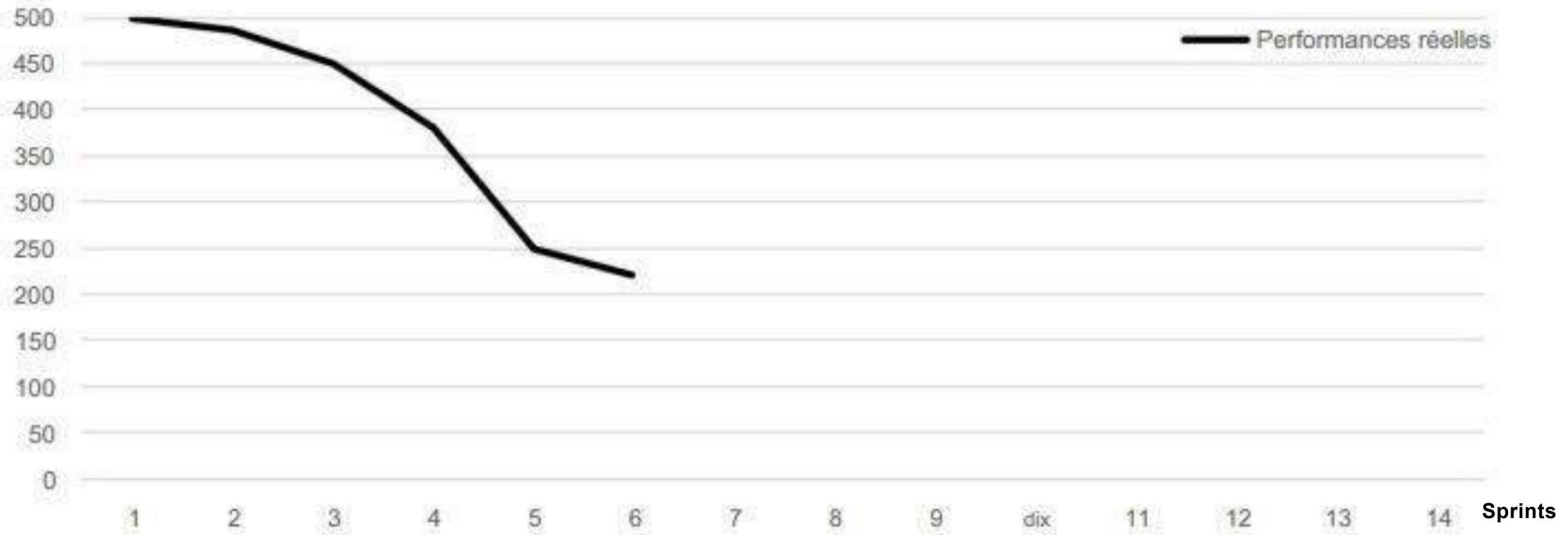
ARTEFACT 5 : SURVEILLER LES PROGÈS

- Burndown chart afin de visualiser l'avancée du développement lors d'un Sprint.
- Diagramme d'avancement du projet.
- Information "Radiateurs" = big visible chart
- Burndown Bars
- Burnup charts
- Diagramme de flux cumulatifs
- Calendrier Niko-Niko

ARTEFACT 5 : SURVEILLER LES PROGÈS

- Burndown chart

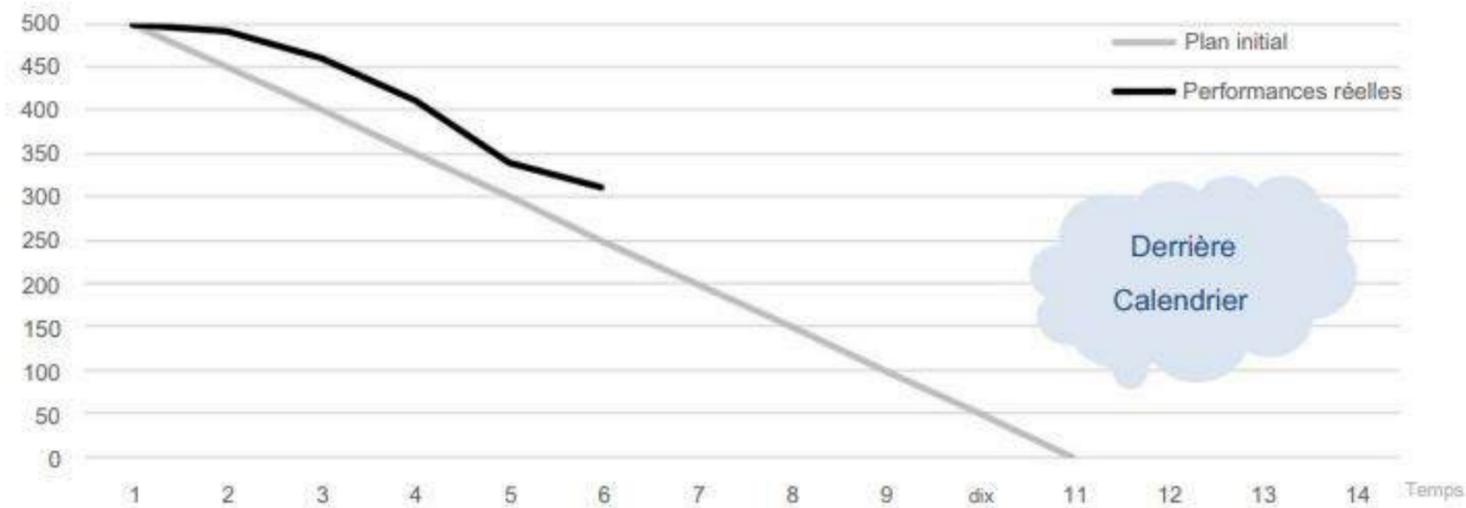
Story points restants



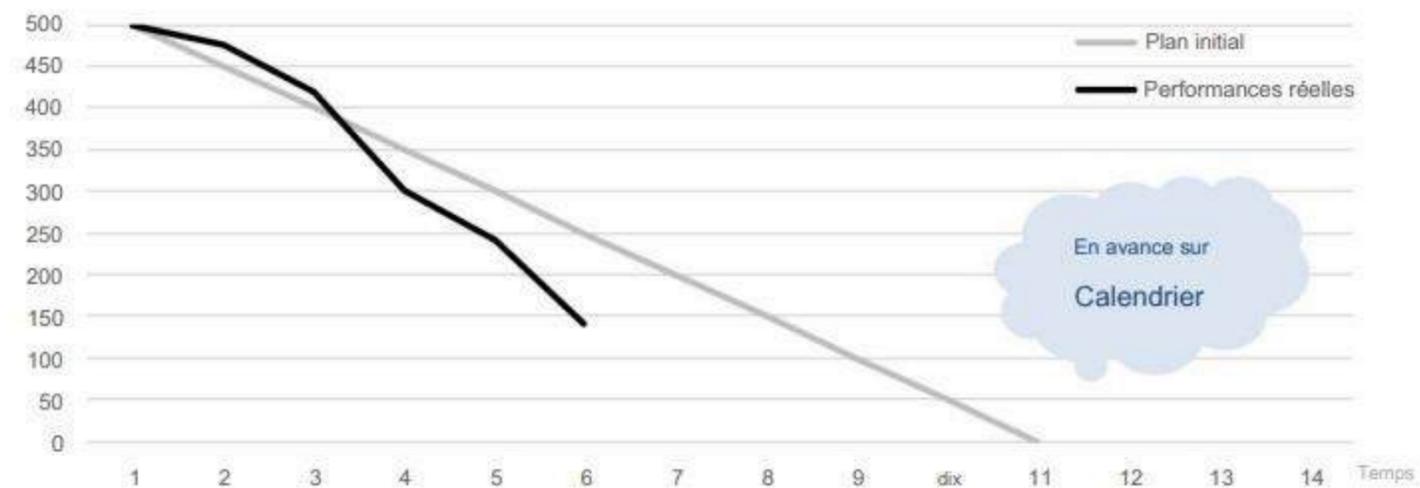
ARTEFACT 5 : SURVEILLER LES PROGÈS

- Burndown chart

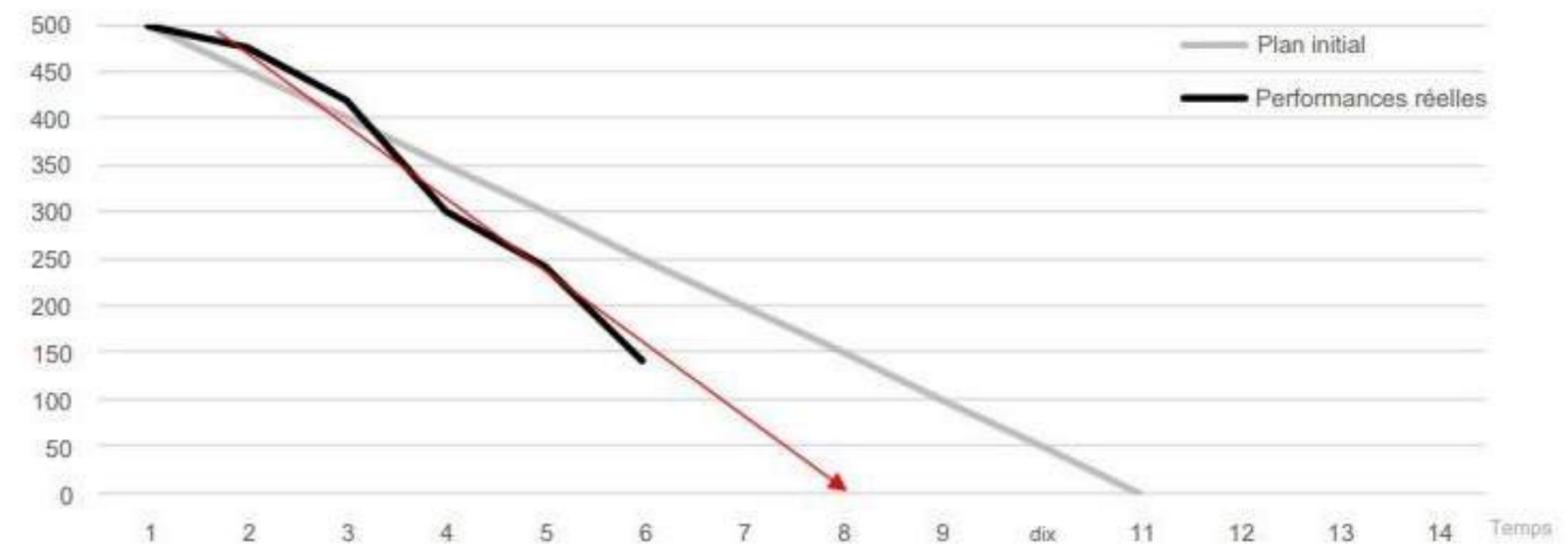
Points d'histoire restants



Points d'histoire restants



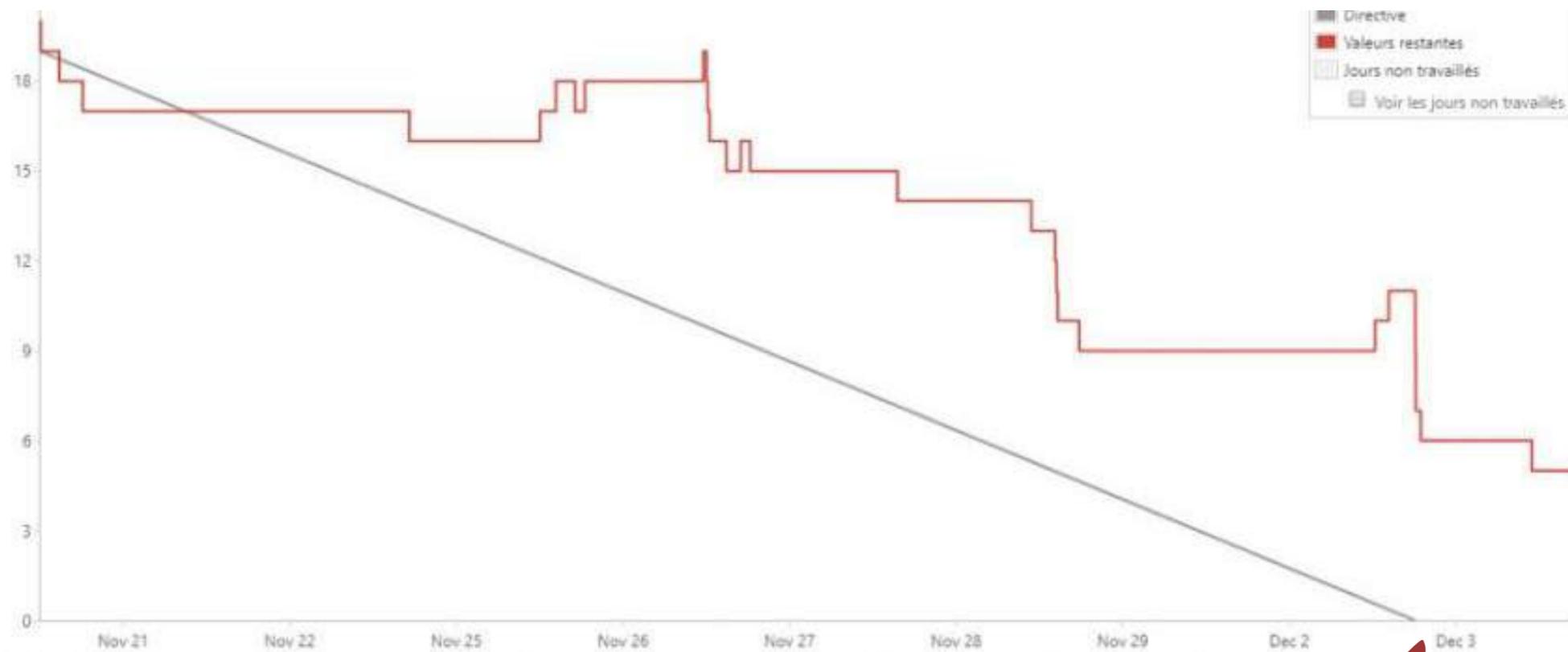
Points d'histoire restants



Le projet sera fait en 8 sprints

ARTEFACT 5 : SURVEILLER LES PROGÈS

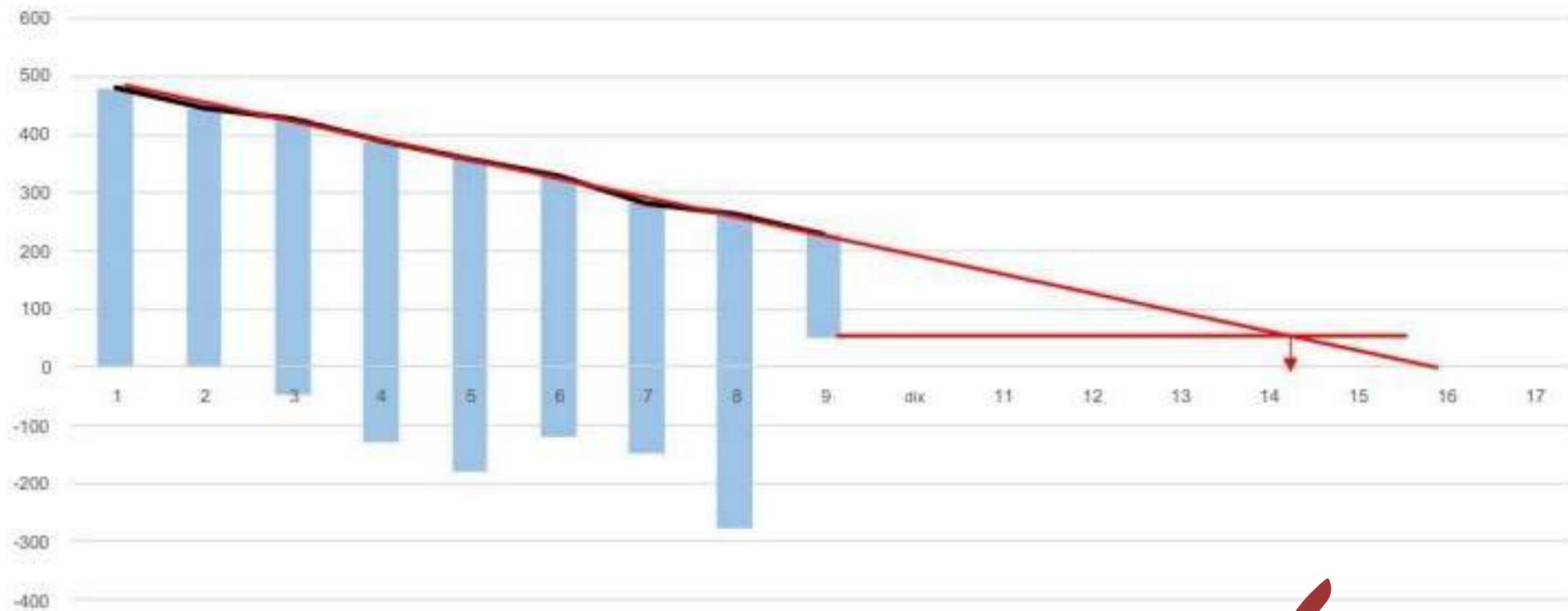
- Burndown chart



L'objectif du sprint n'est pas atteint, l'équipe de développement n'a pas su respecter son engagement.

ARTEFACT 5 : SURVEILLER LES PROGÈS

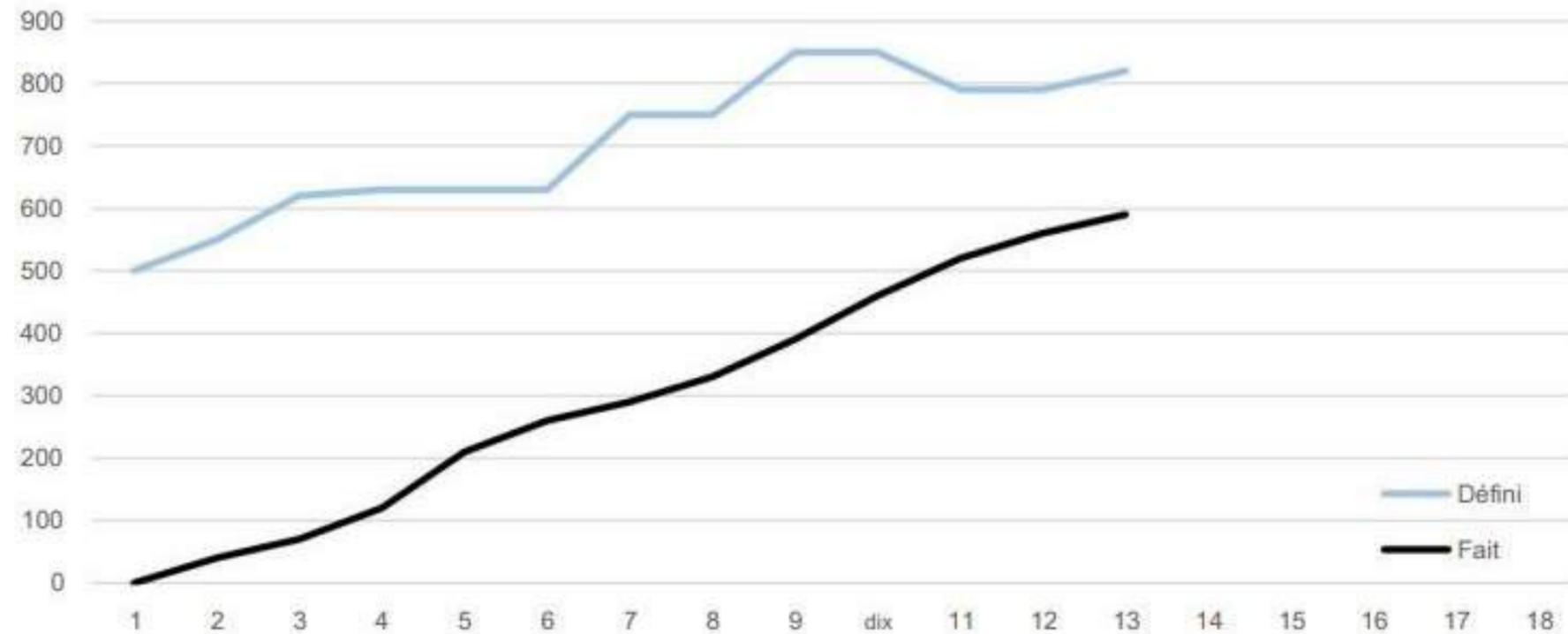
- Burndown bar



Le projet sera fait en 14 sprints

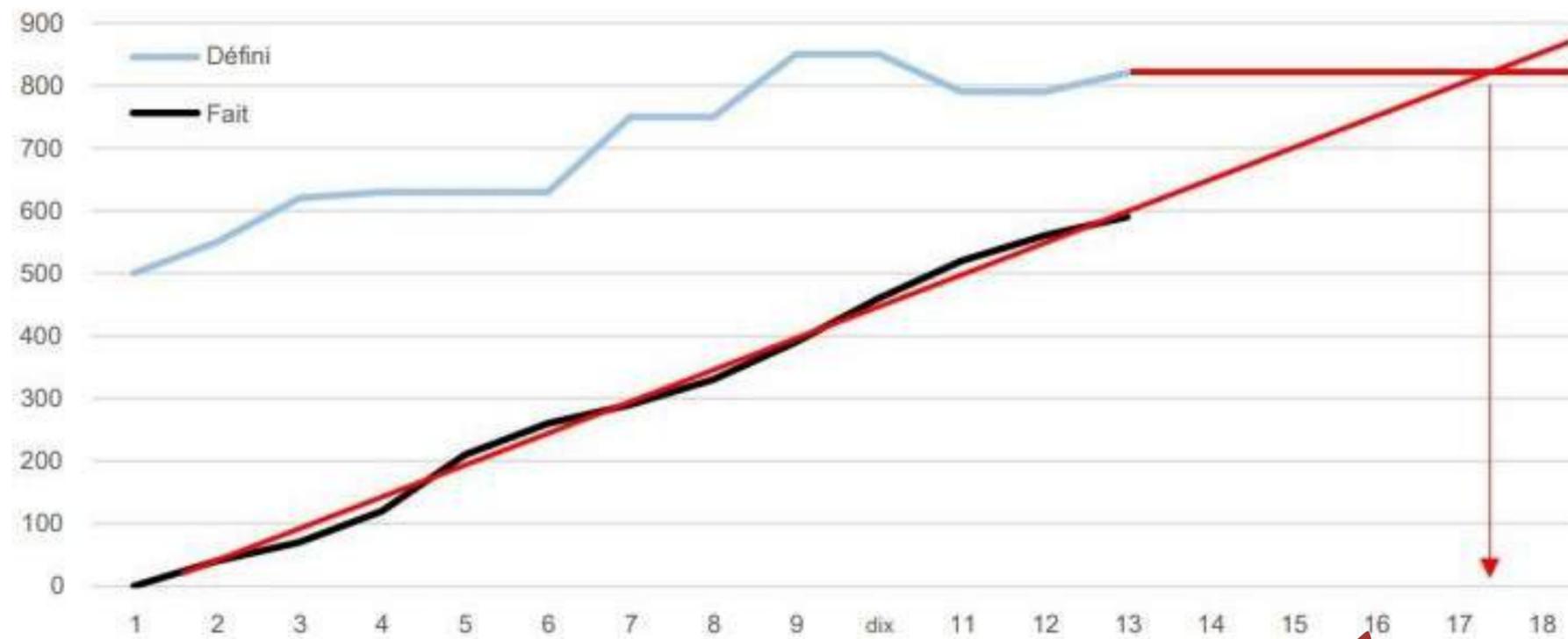
ARTEFACT 5 : SURVEILLER LES PROGÈS

- Diagramme de combustion



ARTEFACT 5 : SURVEILLER LES PROGÈS

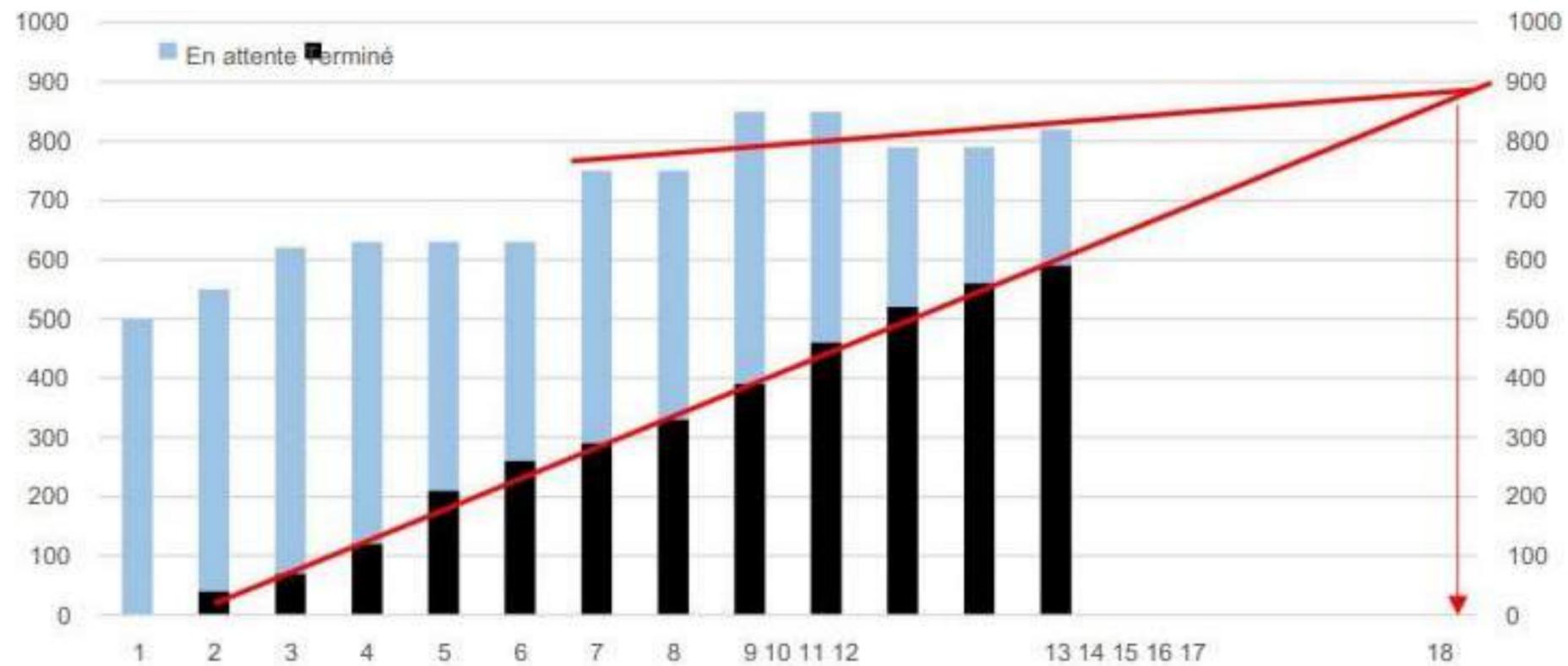
- Diagramme de combustion



Le projet sera fait en 17 sprints

ARTEFACT 5 : SURVEILLER LES PROGÈS

- Diagramme de flux cumulatif



ARTEFACT 5 : SURVEILLER LES PROGÈS

- Calendrier Niko-Niko

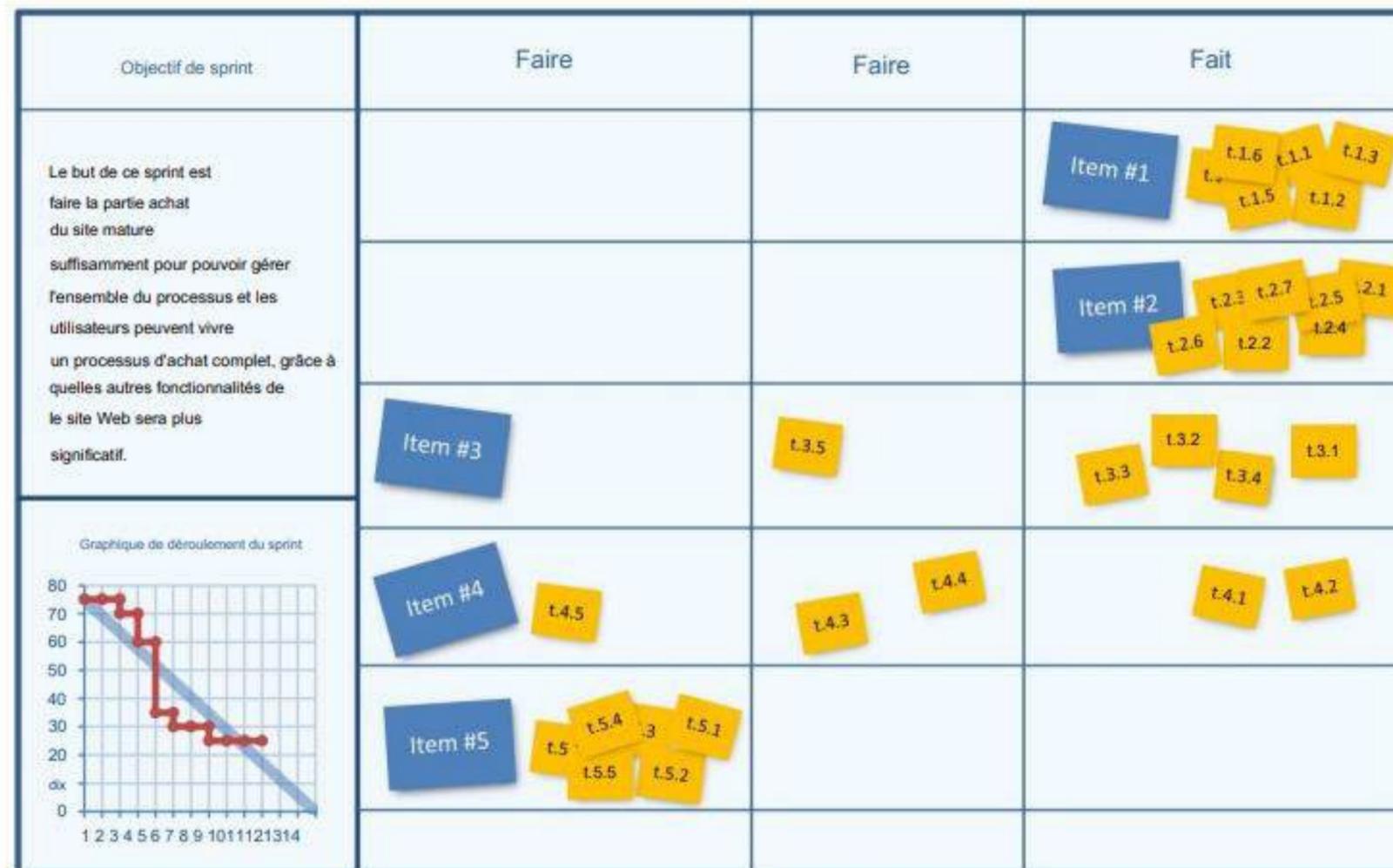
	MON	TUE	WED	THU	FRI
MATT					
SHARON					
WOJCIECH					
SURYA					
ARUN					

ARTEFACT 6 : SURVEILLANCE DE LA PROGRESSION DU SPRINT

- Daily Scrum
- Scrum à l'échelle
- Scrum of Scrums
- Synchronisation
- Diviser les équipes
- Types de contrats et Scrum :
 - Temps et moyens ou unité fixe
 - Prix fixe

ARTEFACT 6 : SURVEILLANCE DE LA PROGRESSION DU SPRINT

Daily Scrum - suivi de la progression du sprint



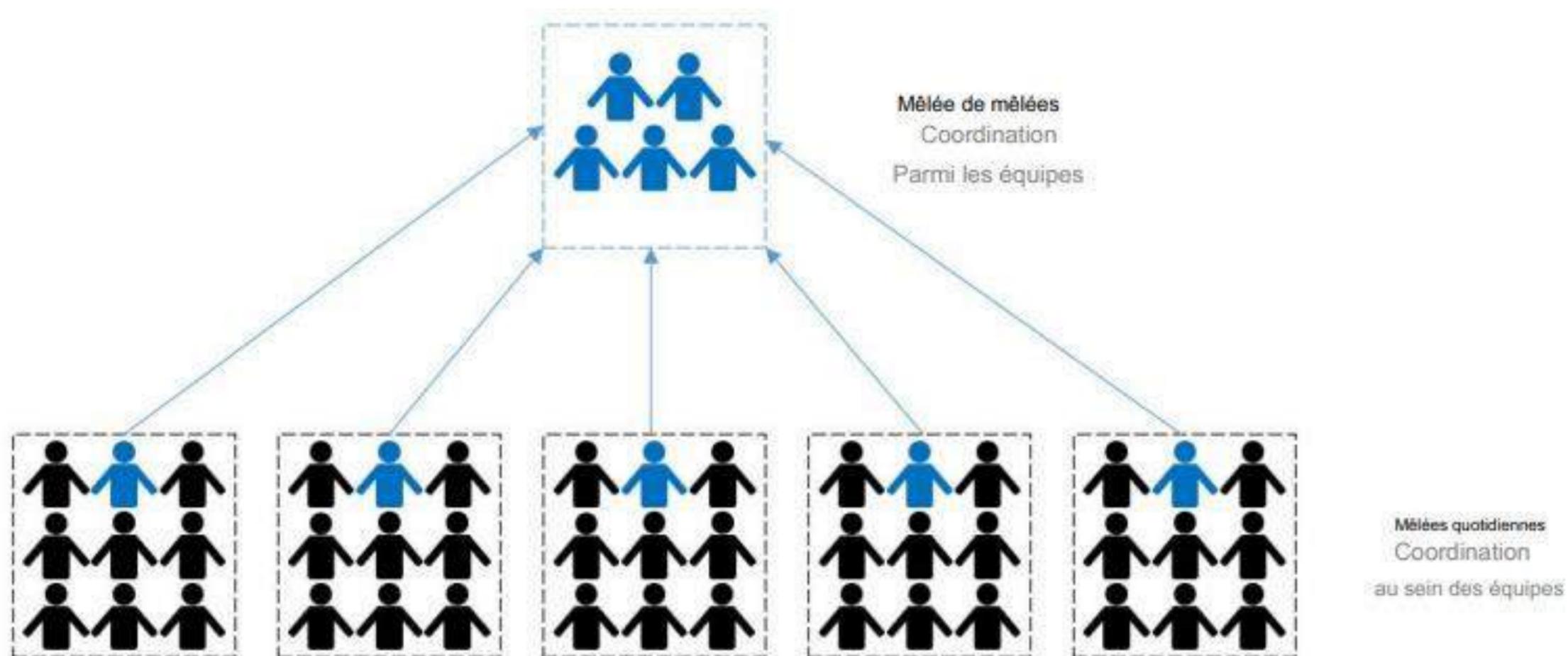
ARTEFACT 6 : SURVEILLANCE DE LA PROGRESSION DU SPRINT

Scrum à l'échelle

- Cadre permettant d'utiliser plusieurs scrum team
- Roles :
 - Chief Scrum Master (Peut travailler avec des SM locaux)
 - Chief product owner (Peut travailler avec des PO locaux)
 - Équipes de développements (max 9)
- Ideal pour des projets complexes

ARTEFACT 6 : SURVEILLANCE DE LA PROGRESSION DU SPRINT

Scrum of scrums



ARTEFACT 6 : SURVEILLANCE DE LA PROGRESSION DU SPRINT

Synchronisation

- Préférable de faire les réunions de synchro (daily scrum) de toutes les équipes en même temps

ARTEFACT 6 : SURVEILLANCE DE LA PROGRESSION DU SPRINT

Types de contrats

- **Temps et moyens ou unité fixe** : Facture le client **soit en fonction des heures de travail, soit par Sprint** (sur la base d'une composition d'équipe et d'une durée fixes pour les Sprints). **Mieux pour Agile.**
- **Prix fixe** : tous les contrats au forfait reposent sur une définition initiale du périmètre, qui est prédictive plutôt qu'adaptative. Ce n'est donc **pas vraiment Agile.**

CONDITIONS PRÉALABLES À SCRUM

1. Une équipe Agile
2. Un produit capable d'être produit de manière incrémentale et itérative
3. Une entreprise Agile
4. Un client Agile



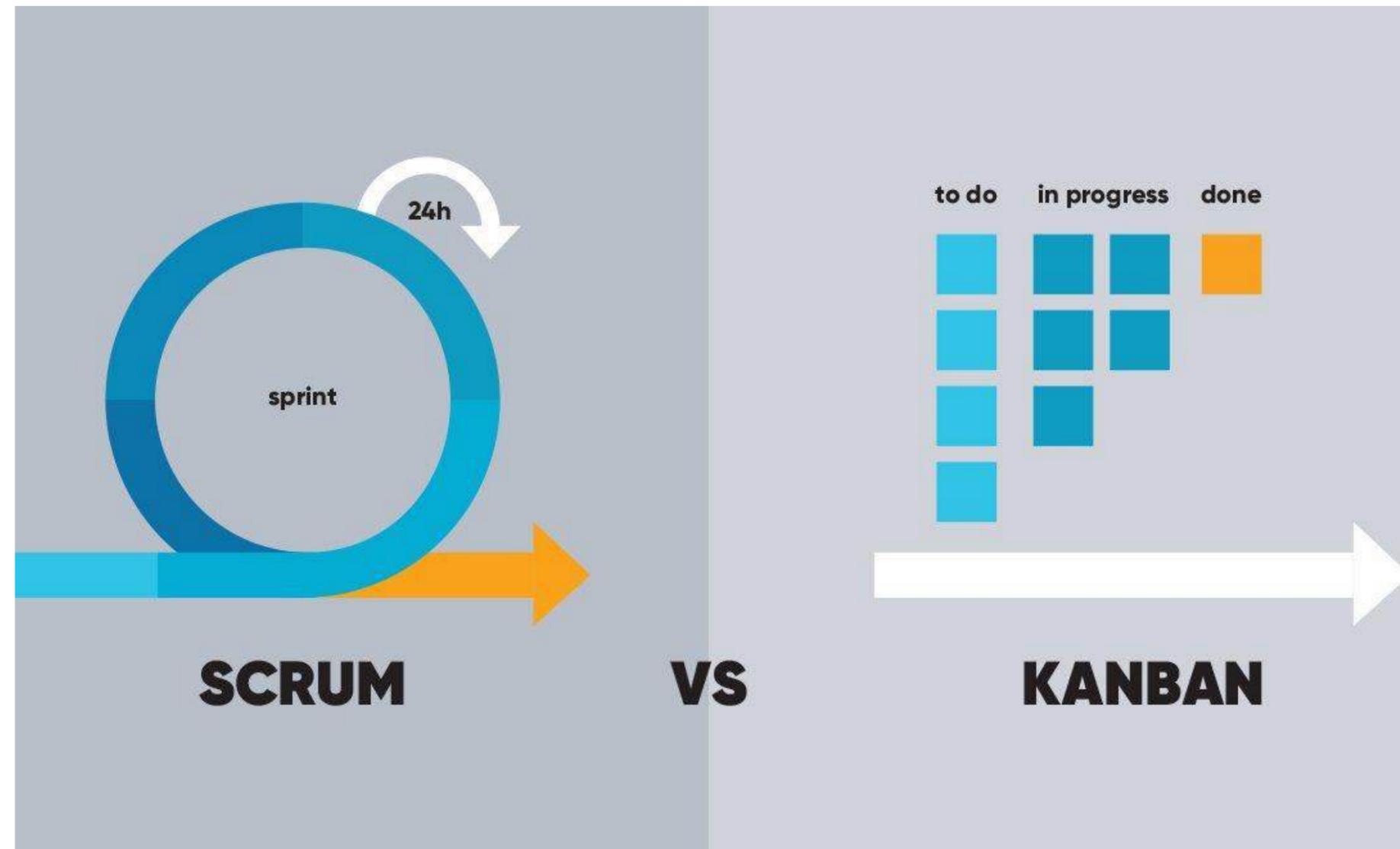
Kanban et ScrumBan

KANBAN

Technique pour gestion de projet

3 règles :

- Le travail doit être visualisé
- Les travaux en cours (WiP) doivent être limités
- Le travail doit être tiré plutôt que poussé



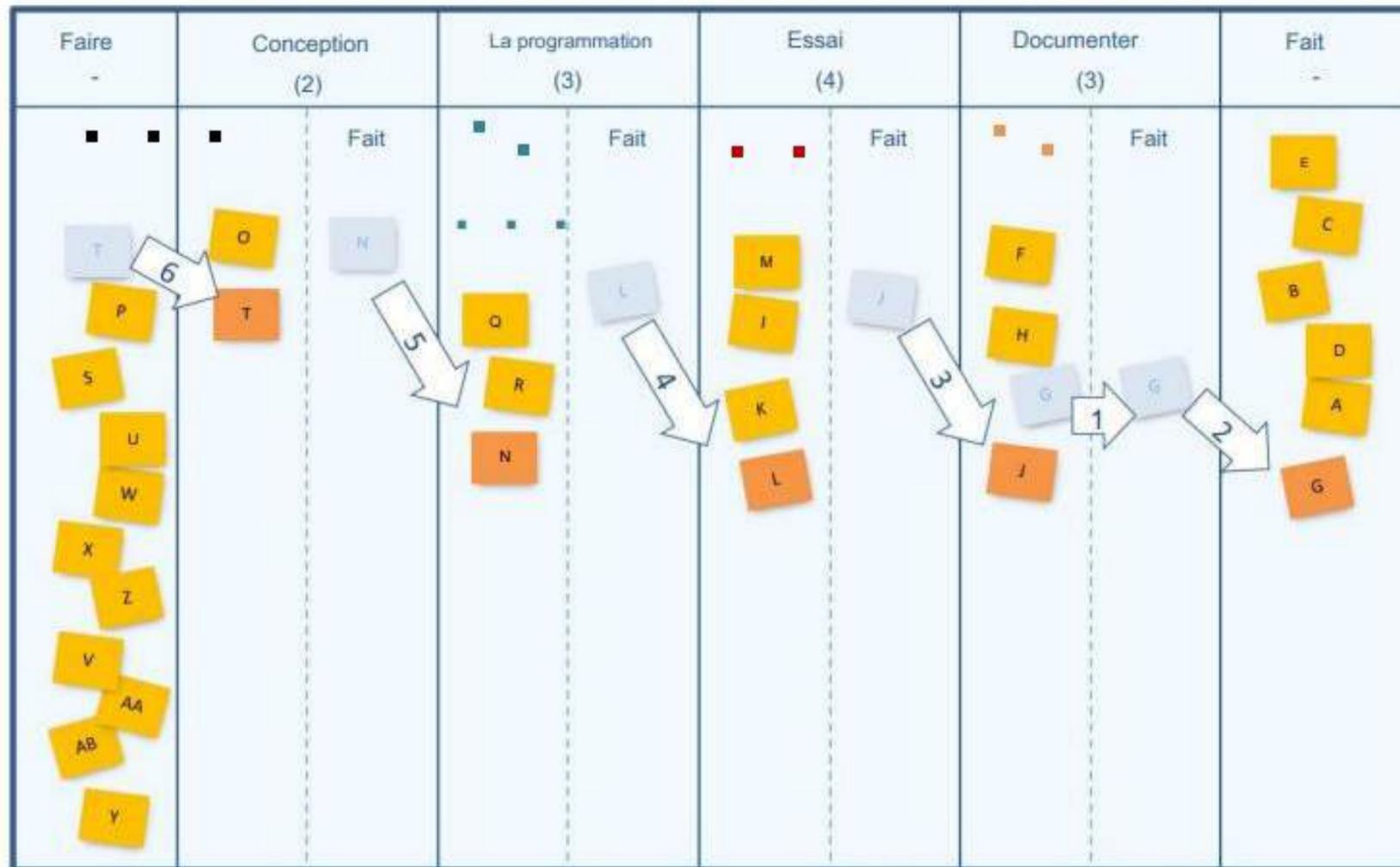
KANBAN

Exemple tableau Kanban

Faire -	Conception (2)	La programmation (3)	Essai (4)	Documenter (3)	Fait -
■ ■ T P S U W X Z V AA AB Y	■ O Fait N	■ ■ ■ Q R L	■ ■ M I K J	■ ■ F H G	E C B D A

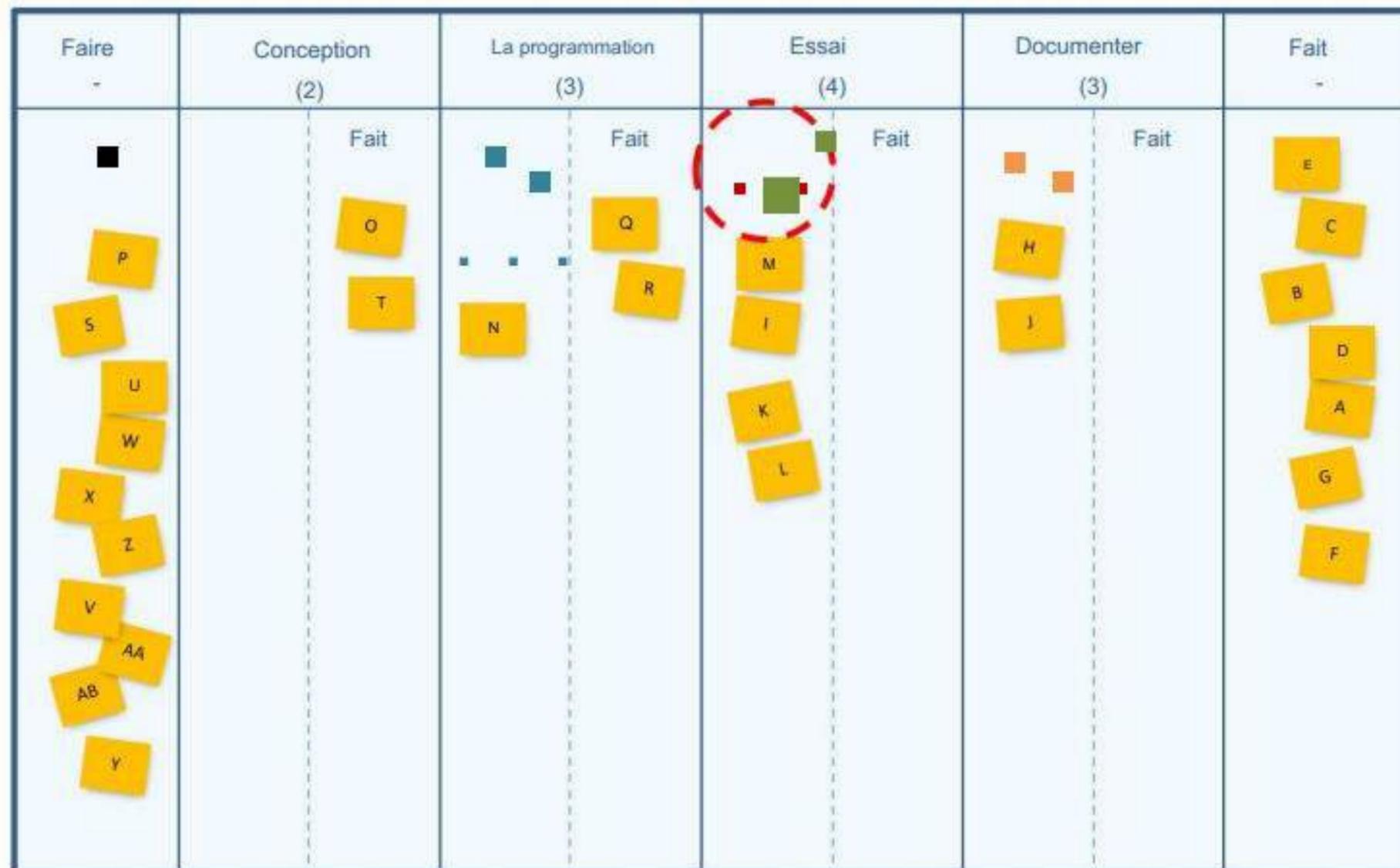
KANBAN

Dynamique Kanban



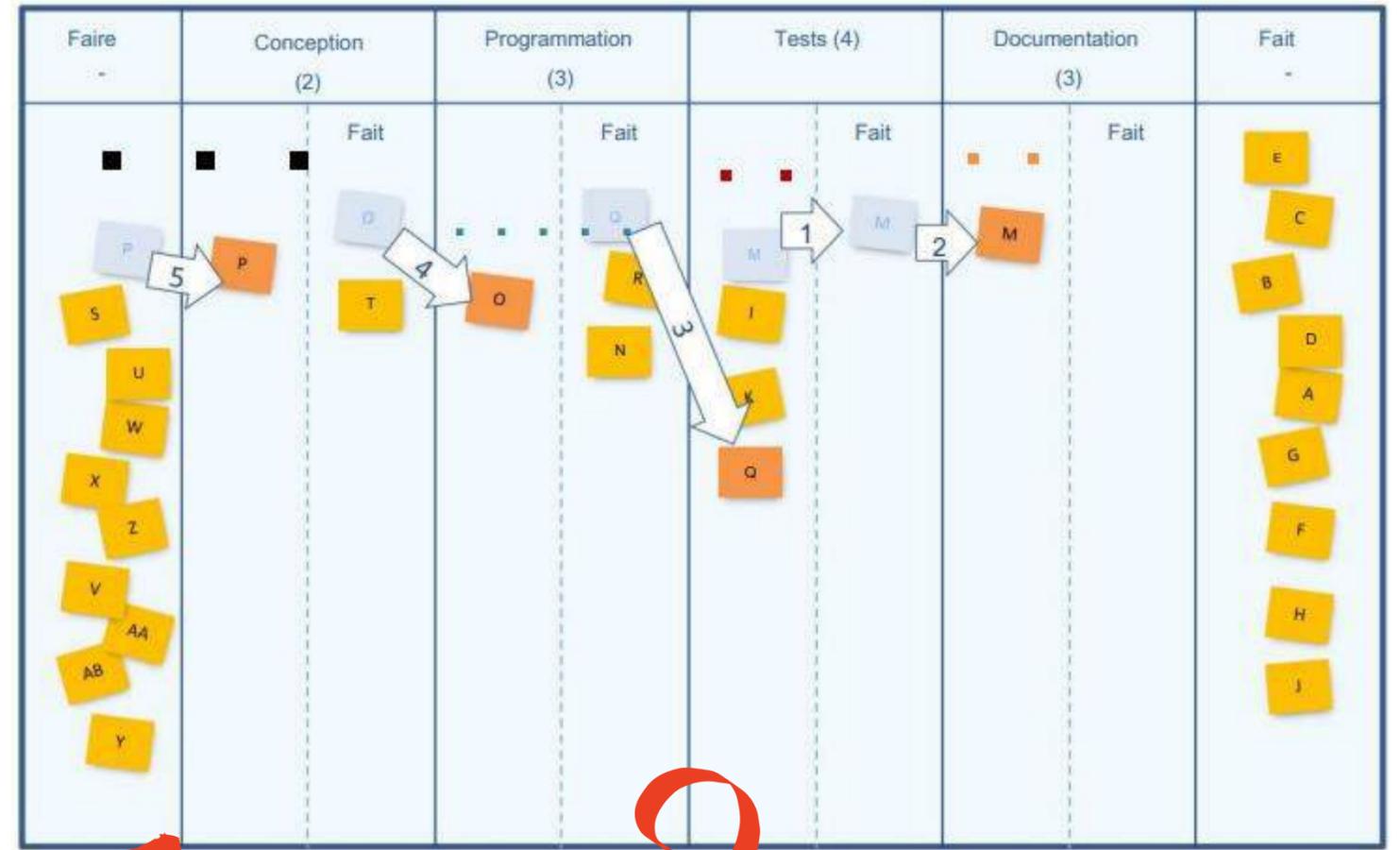
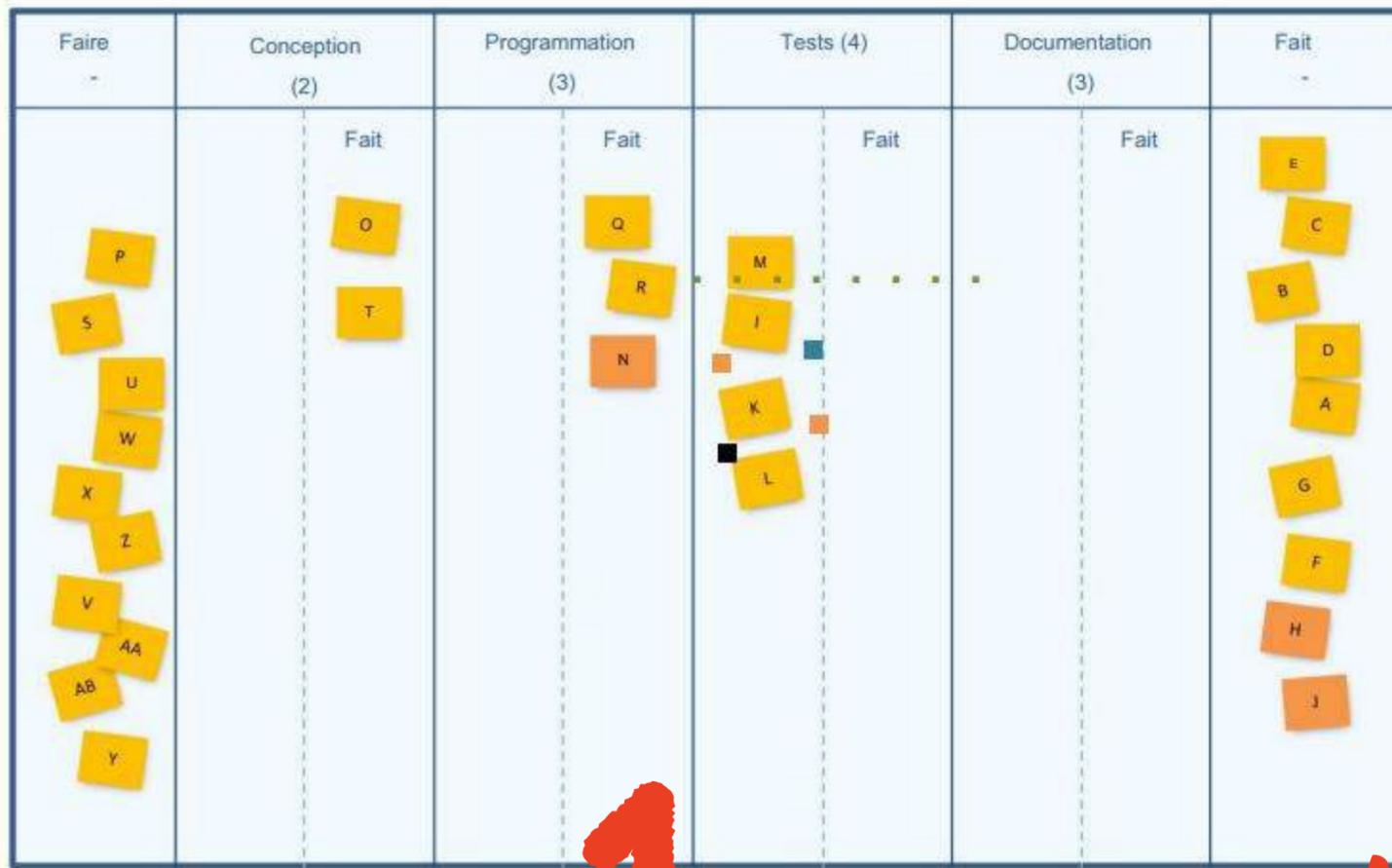
KANBAN

Que fait-on en cas de blocage ?



KANBAN

Que fait-on en cas de blocage ?





SCRUMBAN

Être Agile, c'est s'adapter, plutôt que d'adopter un certain ensemble de termes.

Presque tout est obligatoire dans Scrum et vous ne pouvez omettre aucun aspect. Mais certaines personnes le font ! Par exemple:

- *Nous utilisons Scrum, mais nous ne gardons pas la timebox des Sprints.*
- *Nous utilisons Scrum, mais nous définissons la durée du Sprint dans le Sprint Planning.*
- *Nous utilisons Scrum, mais nous ne laissons pas le Product Backlog évoluer.*
- *Nous utilisons Scrum, mais nous ne jugeons pas nécessaire d'avoir des rétrospectives Sprint*

SCRUMBAN - PARTICULARITÉS

- Réunion de planification (Sprint Planning) : nous n'en avons plus besoin.
- Réunion de bilan (Sprint Review) : elle reste nécessaire et indispensable, car le framework est néanmoins adaptatif et nous avons besoin des commentaires des clients. Les réunions de révision se déroulent comme suit :
 - Soit après un certain volume de travail (par exemple après avoir développé chaque 100 user points ou
 - À certains intervalles (toutes les trois semaines)
- Réunion rétrospective (Sprint Retrospective) : elle est toujours nécessaire, et elle se tiendrait après chaque réunion d'examen.
- Standup quotidien (Daily Scrum) : il est toujours obligatoire et se fait quotidiennement.

SCRUMBAN - PARTICULARITÉS

L'état des artefacts :

- Product Backlog : exactement le même que le Product Backlog Scrum. Elle est visualisée sous la forme de la colonne To Do dans le tableau Kanban.
- Sprint Backlog : nous ne l'avons plus.
- Incrément : nous avons toujours la même définition des incréments et juste avant chaque réunion d'examen.
- Définition de Done : exactement la même que celle du Scrum DoD.
- Suivi des progrès vers un objectif : cette mesure du projet se fait exactement comme nous le faisons dans Scrum.
- Suivi de l'avancement du Sprint : nous ne l'avons plus.

SCRUMBAN

Scrum	Kanban	Scrumban
<p>✗ Continuous flow</p>	<p>✓ Continuous flow</p>	<p>✓ Continuous flow</p>
<p>✓ Team Roles</p>	<p>✗ Team Roles</p>	<p>✗ Team Roles</p>
<p>✗ Easy To Adapt</p>	<p>✓ Easy To Adapt</p>	<p>✗ Easy To Adapt</p>
<p>✗ Reactive to Changes</p>	<p>✓ Reactive to Changes</p>	<p>✓ Reactive to Changes</p>

VIDEOS ET OUTILS UTILES

Guide Scrum expliqué :

<https://www.youtube.com/watch?v=XOrQH8UVTJM&pp=ygUebGVzIGV2ZW5lbWVudHMgc2NydW0gZXhwbGlxdCOp>

Cours complet Scrum :

https://www.youtube.com/watch?v=SyIN_YMfoQs&pp=ugMICgJmchABGAHKBQ5hcnRlZmFjdCBzY3J1bQ%3D%3D

Simulation d'examen

Syllabus et guide

CONTACT :

INFO@LETECHSPACE.COM

WWW.LETECHSPACE.COM



/letechspace



/letechspace

TECHSPACE BRUXELLES :

Cantersteen 12, 1000 Bruxelles (BeCentral, Gare Centrale)

TECHSPACE WALLONIE :

Esplanade Simone Veil 1, 4000 Liège (Silversquare Guillemins)