

Méthodes agiles de gestion et amorçage de projet

Stéphanie Putaggio, Formatrice Introduction aux Méthodes Agiles | Compétences Numériques Fondamentales
7 et 8 décembre 2020, Simplon Marseille

Déroulé de la formation

7 et 8 décembre :

- 2 jours de formation théorique menés à travers un projet fil rouge de création d'une solution numérique

Trois jours en février :

- 2 jours d'accompagnement autour d'un cahier des charges créés par vos formateurs
- 1 jour de coaching pour vous préparer à passer la certification

Certification

Préparation d'une rétrospective à présenter devant le jury et les faire participer à cette rétrospective.

L'apprenant devra présenter cette rétrospective dans le cadre d'un projet web poussé, mis en œuvre avec les méthodes agiles.

L'objectif de cette rétrospective est de restituer la gestion du projet fil rouge en groupe, mis en place lors de la formation, afin de permettre au jury de juger de la bonne compréhension individuelle des outils.

Ce que l'apprenant.e doit présenter lors de sa rétrospective :

- récapitulatif des outils et du matériel utilisé
- rassemblement des données : quels sont les événements qui se sont passés, quelles avancées, quelles décisions, quels défis ?
- présentation de la philosophie d'utilisation des outils Agiles, le déroulement de leur application et leur capitalisation
- discussion constructive avec le jury après cette présentation, l'apprenant.e cherche à générer des avis
- clôture: invention et présentation des actions d'amélioration (2 ou 3 axes d'amélioration, max), listing des principales frustrations , synthèse.

Installation du candidat (5 minutes)

Epreuve (25 minutes) :

- 5 min de présentation du candidat devant jury
- 20 minutes d'échanges avec le jury

Délibération du jury - en l'absence du candidat (10 minutes)

Que signifie « Agile » en gestion de projet ?

Méthodes Agiles vs méthodes classiques

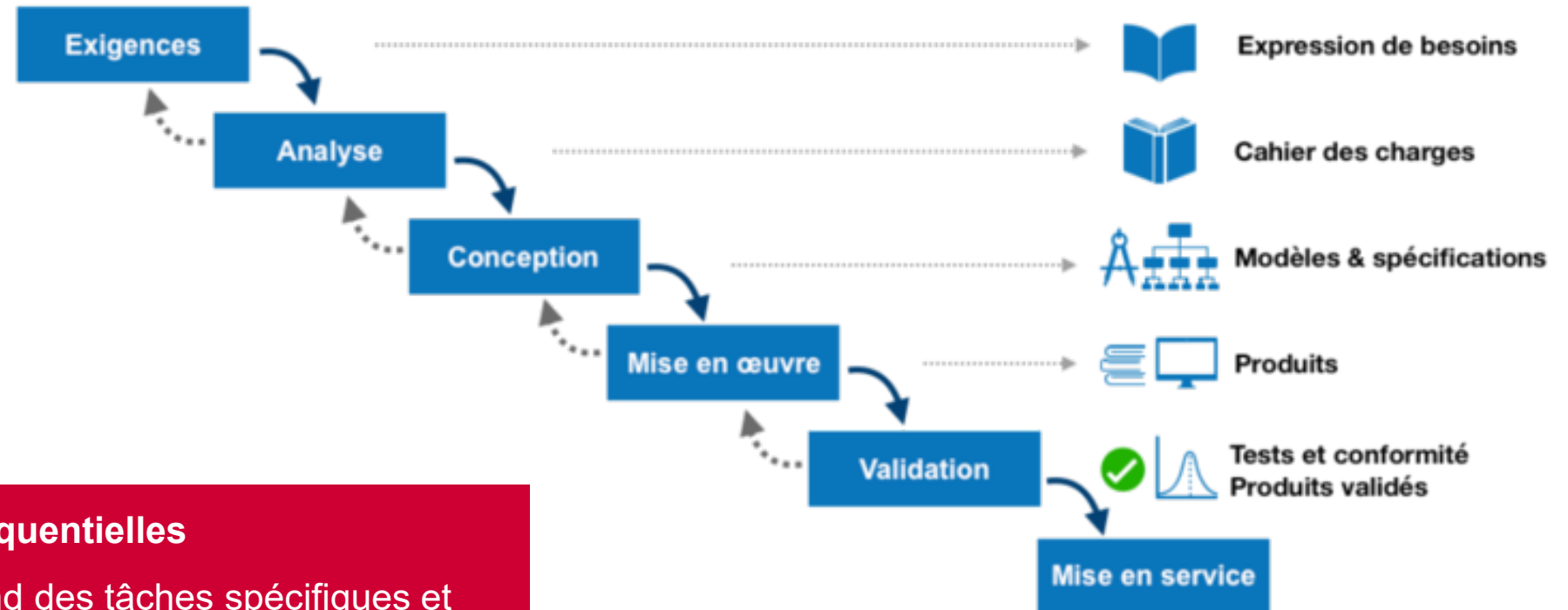
Début

Des étapes

Fin



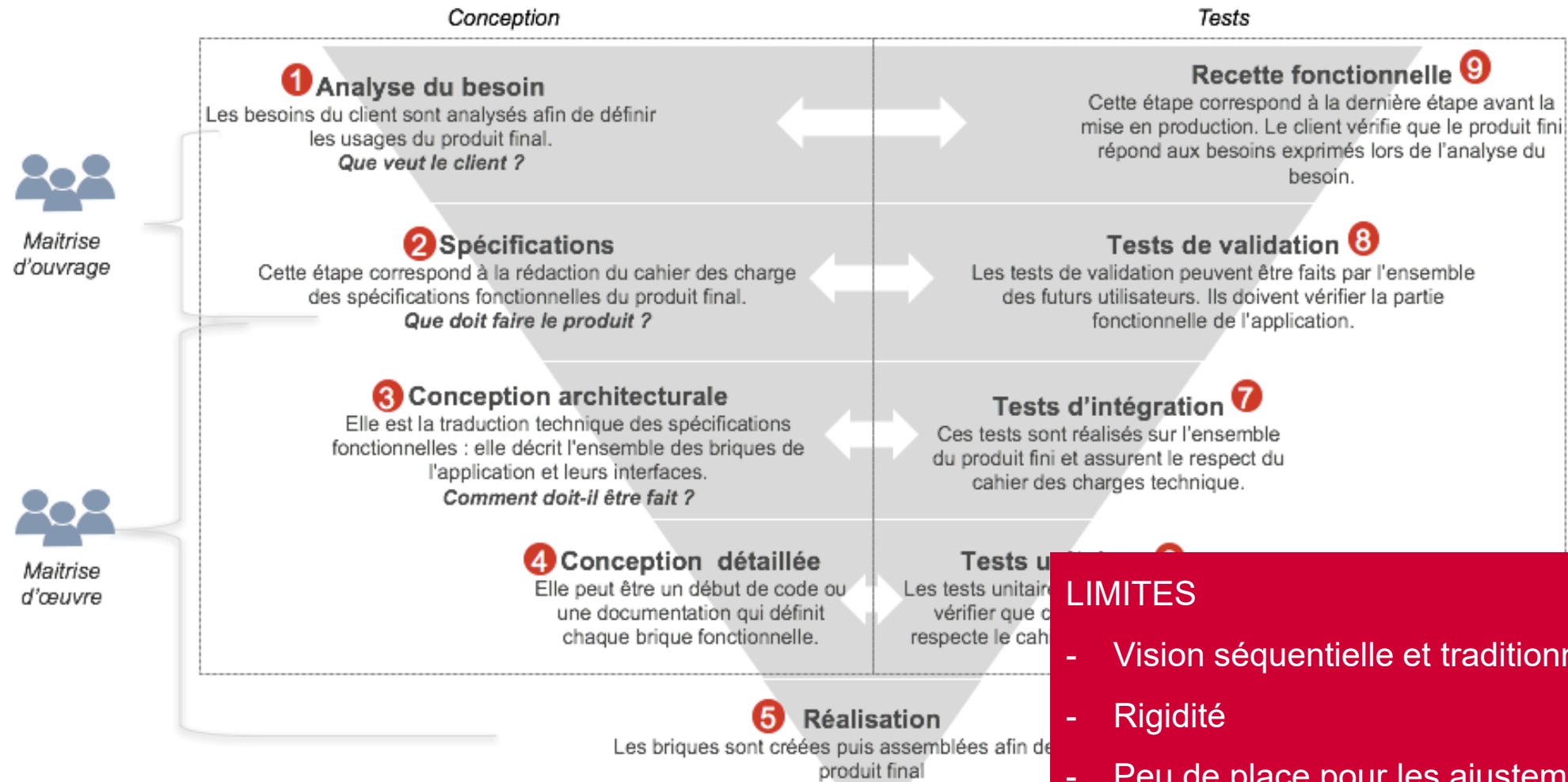
Modèle en cascade



Phases linéaires et séquentielles

Chaque phase comprend des tâches spécifiques et dépend des résultats de la phase précédente

Modèle en V



LIMITES

- Vision séquentielle et traditionnelle
- Rigidité
- Peu de place pour les ajustements et imprévus
- Peu adapté dans un contexte d'évolution rapide

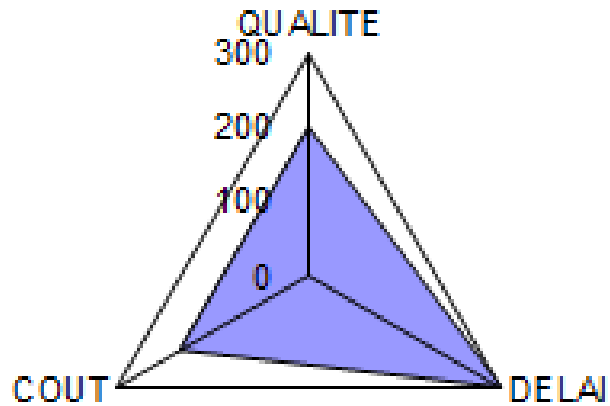
Satisfaction client / utilisateurs

QUALITE

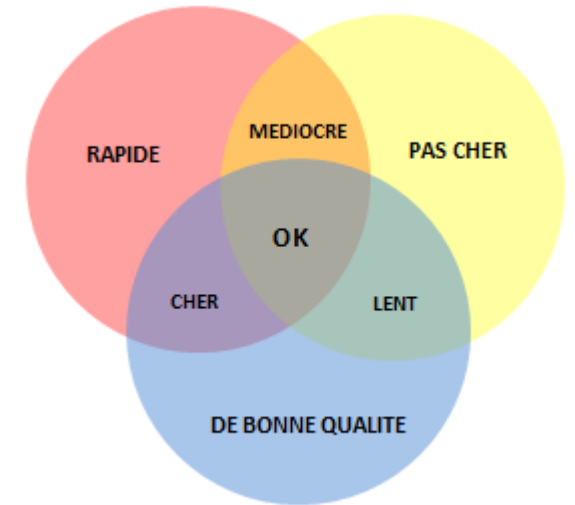
PROJET

COÛTS

DELAIS



Respect du budget défini
au lancement du projet



Réaliser le projet dans le
respect des délais annoncés



**Comment le client
a exprimé son besoin**



**Comment le chef de
projet l'a compris**



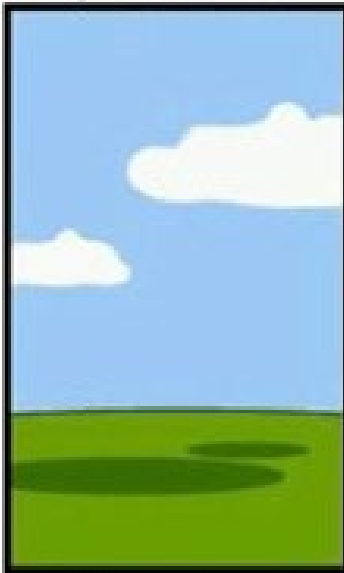
**Comment l'ingénieur
l'a conçu**



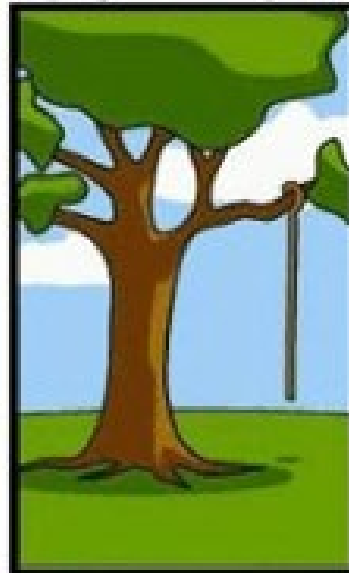
**Comment le
programmeur l'a écrit**



**Comment le responsable
des ventes l'a décrit**



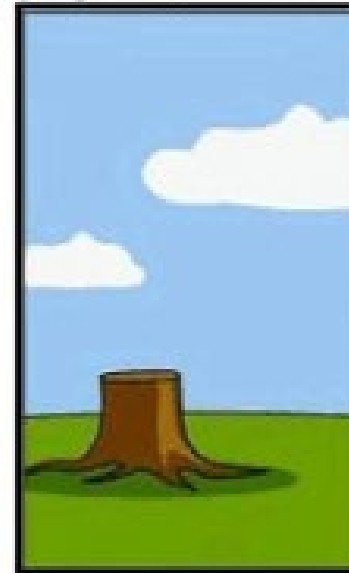
**Comment le projet
a été documenté**



**Ce qui a finalement
été installé**



**Comment le client
a été facturé**



**Comment la hotline
répond aux demandes**



**Ce dont le client avait
réellement besoin**

Manifeste pour le développement Agile de logiciels

Nous découvrons comment mieux développer des logiciels
par la pratique et en aidant les autres à le faire.

Ces expériences nous ont amenés à valoriser :

Les individus et leurs interactions plus que les processus et les outils

Des logiciels opérationnels plus qu'une documentation exhaustive

La collaboration avec les clients plus que la négociation contractuelle

L'adaptation au changement plus que le suivi d'un plan

Nous reconnaissons la valeur des seconds éléments,
mais privilégions les premiers.

Les 4 valeurs Agiles

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

Les 12 principes agiles

Toujours satisfaire le client à travers des livraisons rapides et continues

Bien accueillir tous les changements même les tardifs

Livrer fréquemment un système fonctionnel

Les clients et les développeurs doivent collaborer

Conduire le projet autour d'équipes motivées

La meilleure méthode de faire circuler l'information c'est le contact direct entre collaborateurs

La première mesure d'avancement c'est un logiciel fonctionnel

Le développement doit être durable et à un rythme constant

La bonne conception et l'excellence technique augmentent l'agilité

Simplifier au maximum

Les meilleurs architectures, besoins et conceptions proviennent d'équipes qui s'organisent d'elles-mêmes

L'équipe s'améliore d'une manière autonome et régulière

Avantages

Flexibilité

Les changements et les imprévus sont pris en compte et l'équipe peut réagir rapidement

Implication du client / adhésion de l'équipe

Relation de confiance, moins de perte de temps

Réelle visibilité sur l'avancement du projet

Contrôle qualité permanent

et limites

L'implication du client

Le client doit être disponible et s'intéresser pleinement à son projet

Peu adaptées à une structure hiérarchique forte

Collaboratif

Difficulté pour avoir une vision globale des coûts

Le client doit être prêt à s'adapter

Les individus et les interactions plus que les processus et les outils

Des méthodes

- Scrum
- Kanban
- Lean Startup
- Design Thinking

Extrem Programming (XP)

FDD (Feature-Driven Development)

Dynamic System Development Method (DSDM)

Adaptive software development (ASD)

Crystal Clear

Iterative

1



2



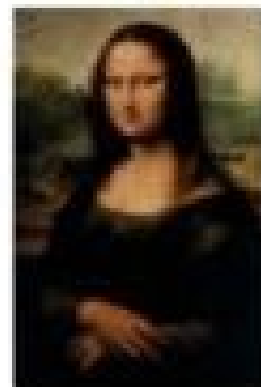
3



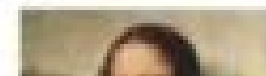
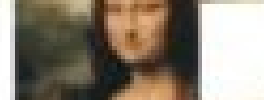
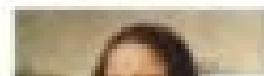
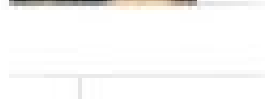
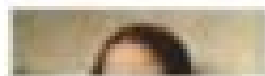
4



5



Incremental



Design Thinking

Démarche de gestion de projet pour apporter des solutions pertinentes à une problématique en partant de l'écoute des besoins des personnes concernées.

Le cœur de la méthode consiste à rencontrer des personnes et à les observer pour comprendre leurs besoins, à fabriquer des prototypes et à les tester pour les améliorer.

Cette démarche mobilise des compétences que nous avons tous naturellement mais que nous avons tendance à sous-exploiter.

Intuition

Intelligence émotionnelle

Créativité

Sens pratique

CENTRÉE SUR
L'HUMAIN

EXPÉRIMENTALE

UNE METHODE

COLLABORATIVE

FONDÉE SUR LA
PRATIQUE

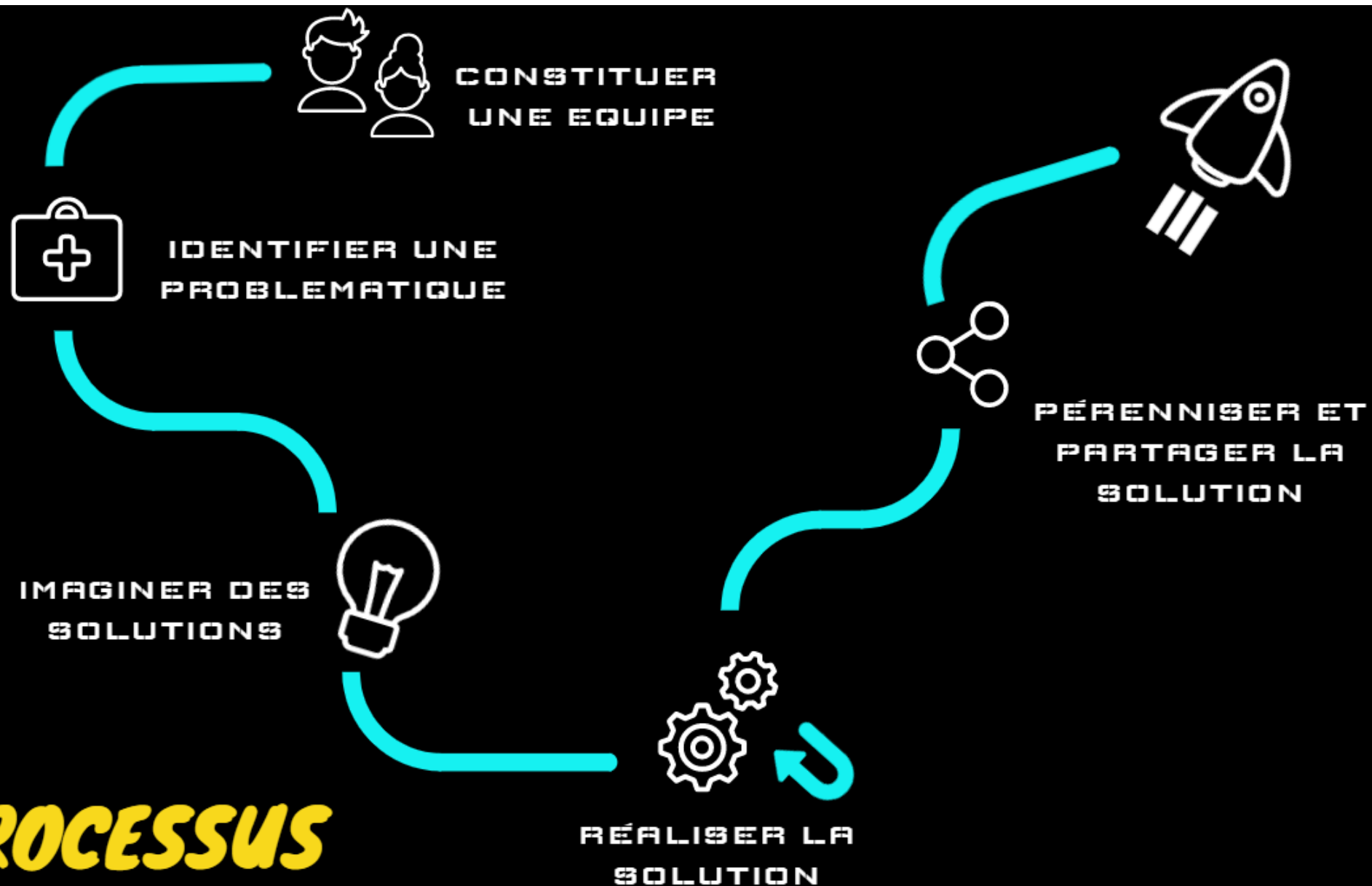
OPTIMISME

UN ETAT D'ESPRIT

LIBRE DE TOUT
PRÉJUGÉ

LÂCHER-PRISE

PROCESSUS



Quelles sont les caractéristiques des entreprises qui « pensent design » efficacement ?

Une culture du prototype : la compréhension que le design ne s'applique pas qu'aux objets

Les géants de l'internet, comme Google, Facebook ou encore Amazon, utilisent le design thinking pour mettre au point et améliorer des plateformes optimisées pour l'utilisateur

Des investissements importants : ces entreprises misent sur l'exploration et l'imagination, et leur offrent des moyens conséquents

Une culture de la synthèse : les firmes les plus innovantes rejettent les projets sur-documentés, et leur préfèrent une pensée forte et synthétique.

L'empathie : ces entreprises innovantes partent des besoins de l'utilisateur, voire de ses frustrations, pour lui fabriquer une expérience mémorable.

Atelier d'innovation et de prototypage rapide



Prénom : Alice
Age : 28 ans
Métier : Fleuriste
Statut : Indépendante

CONTEXTE

- Travaille à mi-temps dans un magasin de fleuriste classique
- Voudrait monter sa propre entreprise de fleuriste
- La concurrence est rude dans ce domaine

SES ENVIES

1. Elle veut se lancer dans un concept original et se différencier de la concurrence
2. Elle veut développer son réseau pour se faire connaître
3. Elle aimerait bien voyager pour son métier et son plaisir
4. Elle veut se tenir informée des innovations et des nouveautés dans son domaine
5. Elle veut développer son concept depuis son domicile sans avoir une boutique

1. Définition de la problématique

Identifier le ou les problèmes majeurs

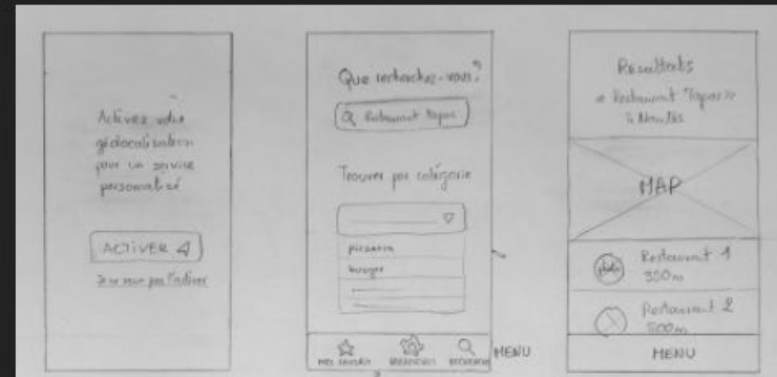
2. Imaginer des solutions

Quelle solution technologique pourriez-vous inventer ?

Création d'une application mobile

3. Réalisation d'un prototype (schéma)

- Faire Simple et clair
- Nom/slogan/logo
- 3 écrans (dessiner des croquis)
- Penser parcours utilisateurs
- Boutons (accueil, retour en arrière...)



4. Conception du prototype



1. Taper le nom de votre projet + Sélection du modèle ✓
2. Dans votre projet, sélection du symbole +
3. Sélectionner *Camera* pour prendre vos 3 croquis en photo ✓
4. Vous pouvez arranger vos photos (rognier, etc.) ✓
5. Sélectionner une de vos photos
6. Créer des liens *add a hotspot* : déplacer le cadre à l'endroit souhaité et ajouter le lien vers la page désirée
7. Une fois vos trois pages terminées, visualiser votre projet ►

5. Tester votre projet auprès des futurs utilisateurs

Pitcher son projet en 60 secondes max

- Le nom de l'appli
- Le problème résolu
- Le fonctionnement et la valeur ajoutée de l'appli

Lean Startup

« Il vaut mieux agir maintenant de manière imparfaite que d'attendre d'être parfait pour agir »

Une méthode managériale pour entrepreneurs et innovateurs utilisée au démarrage d'une activité ou au lancement d'un nouveau produit.

Développée par Eric Ries et résumée dans son ouvrage *Lean Startup : Adoptez l'innovation continue*.

Il est parti de l'observation selon laquelle nombreuses sont les startups qui peinent à décoller et qui échouent.

Il a identifié que l'une des raisons de ces échecs réside dans les ambitions et investissements parfois démesurés comparativement à leur business de départ.

Ces ambitions et ces investissements sont du temps et des ressources gaspillées.

Le cœur du concept se résume en 3 étapes :

Construire

Mesurer

Apprendre

L'idée étant d'obtenir un produit ou service minimum viable (ou MVP en anglais) qui sera rapidement opérationnel et réellement utilisé par les utilisateurs plutôt que de passer un an à concevoir un produit que personne ne voudra.

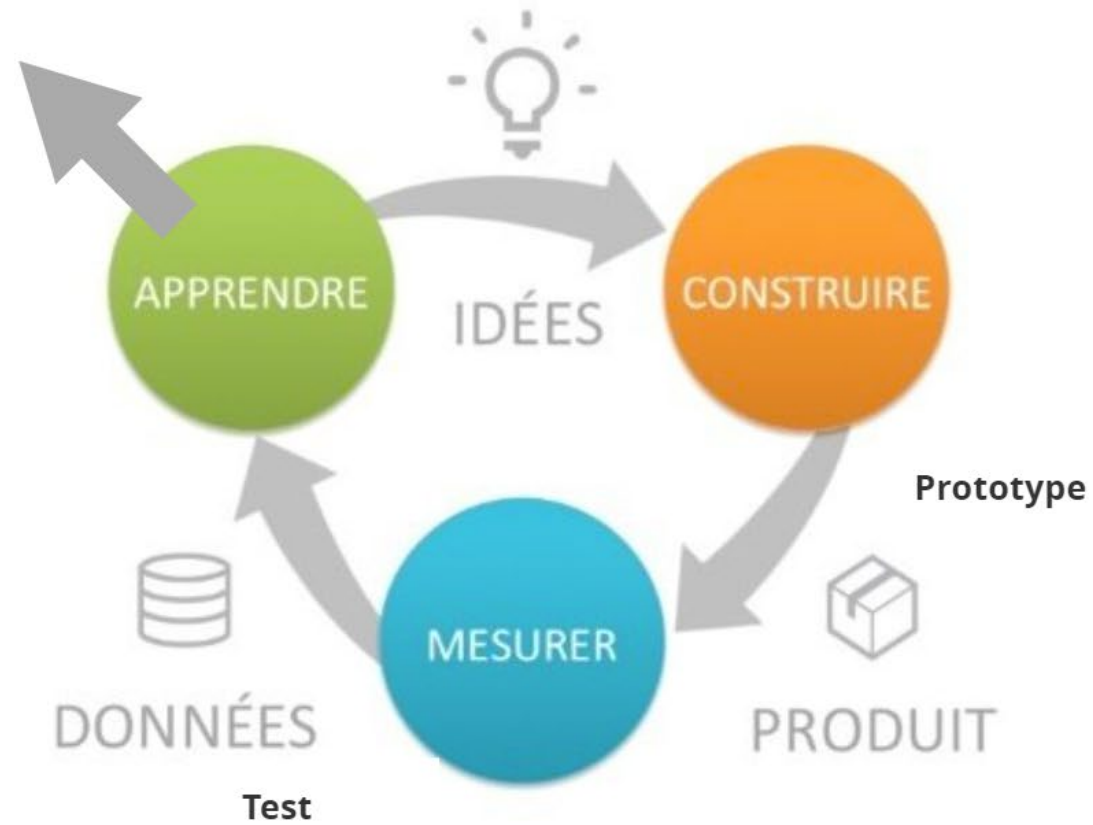
Eric Ries propose une approche permettant à l'entrepreneur de rester maître de son projet et libre de changer de cap rapidement "en testant, en adaptant et en ajustant ses plans" de manière continue.

“Viser grand en commençant petit !”

- Persévérer

- **Pivot** : changements de cibles, de fonctionnalités...

- Abandon



Scrum

Méthode de développement agile orientée projet informatique.

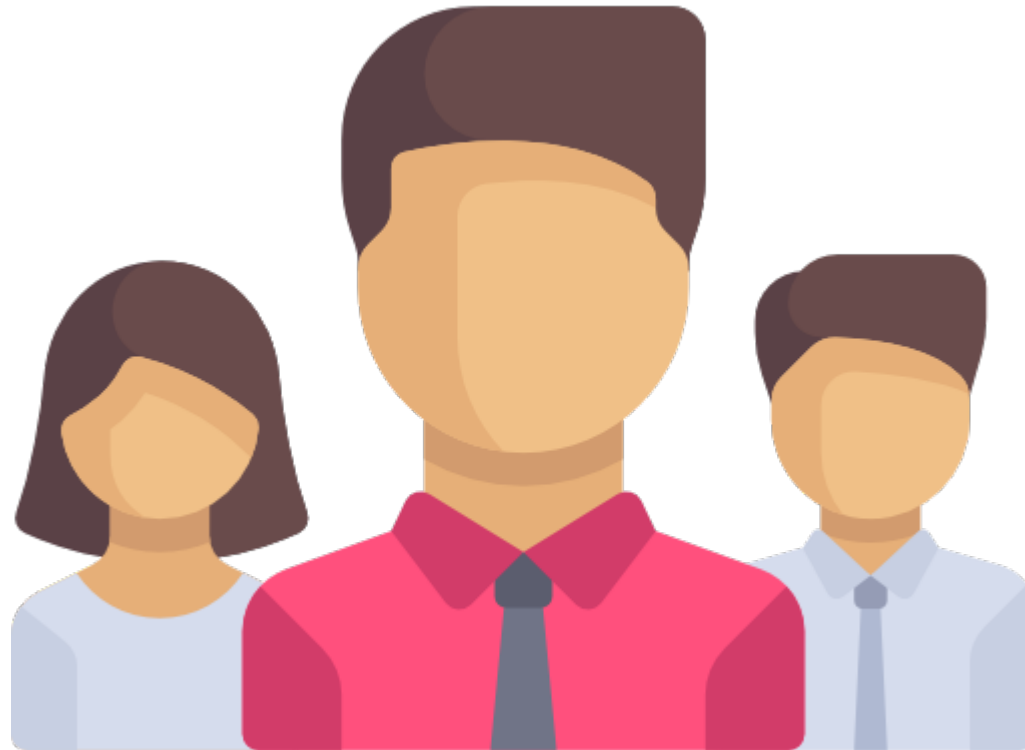
Elle tire son nom du monde du rugby :

- scrum = mêlée

Le principe de base étant d'être toujours prêt à réorienter le projet au fil de son avancement, en impliquant le client tout au long du processus.

Product Owner

Scrum Master



Equipe de dev

Product Owner

Garant de la vision du client

Le PO réalise des choix fonctionnels, en priorisant le développement d'une fonctionnalité au détriment d'une autre. Afin de réaliser ce choix, le PO se base sur 4 caractéristiques (V.R.A.C.) :

- Valeur ajoutée pour les utilisateurs
- Gestion des risques
- Apprentissage (feedback obtenu sur une expérimentation)
- Coûts

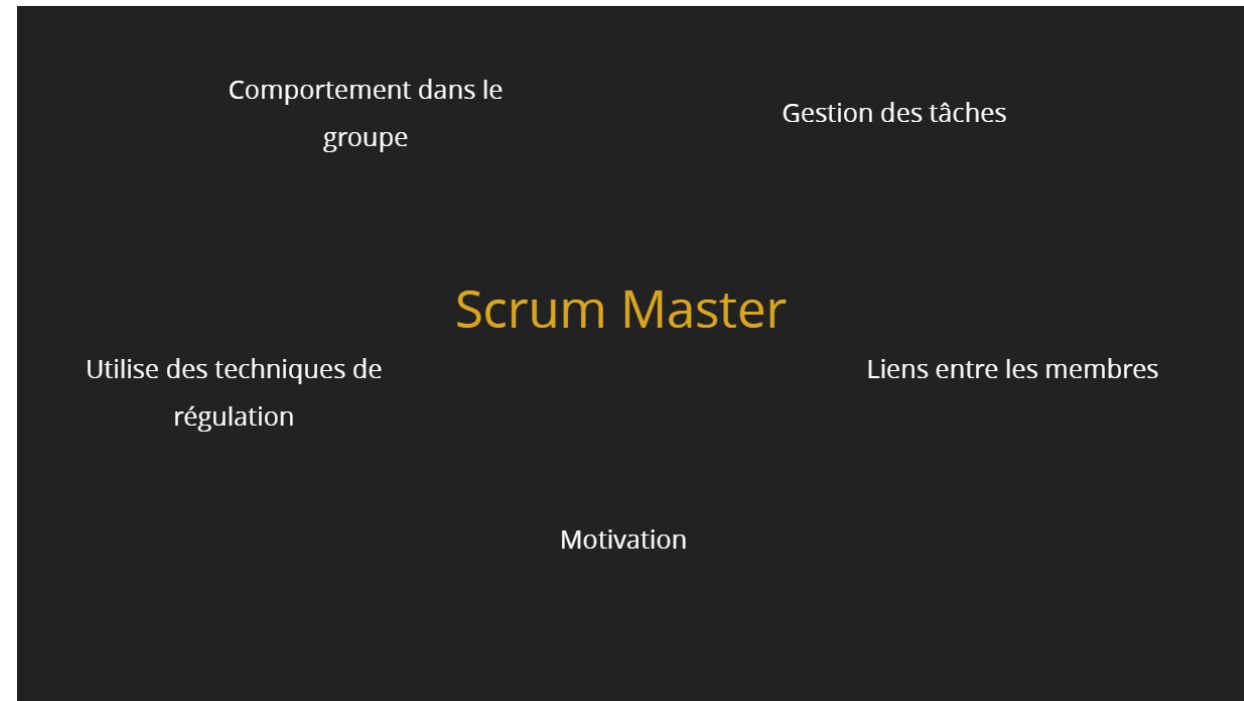
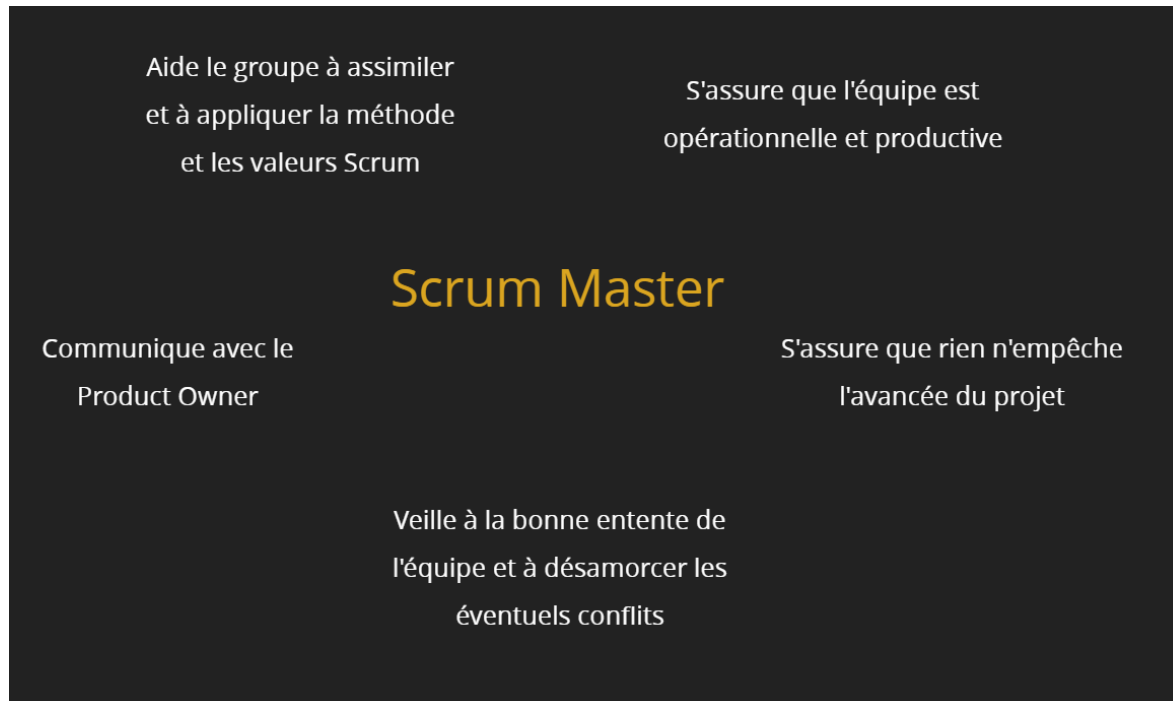


Scrum Master

Véritable facilitateur du projet, le Scrum Master est garant du cadre méthodologique Scrum.

Il veille à conserver une harmonie au sein de l'équipe et à stimuler la créativité.

Il est le garant de l'avancement du projet, grâce notamment à des outils de management visuel



N'est pas un manager

Au service de l'équipe,
facilitateur, coach

Scrum Master

Temp plein

Accompagne l'organisation
dans le changement

Le PO et le SM ne peuvent
pas être la même personne

Equipe de dev

L'équipe de développement va mettre en œuvre les solutions techniques, réaliser les développements.

Idéalement, elle est stable et est composée de trois à dix développeurs.

L'équipe de développement a le pouvoir de s'auto-organiser Elle est pluridisciplinaire et possède toutes les compétences nécessaires pour créer un incrément du produit.

Solidarité d'action

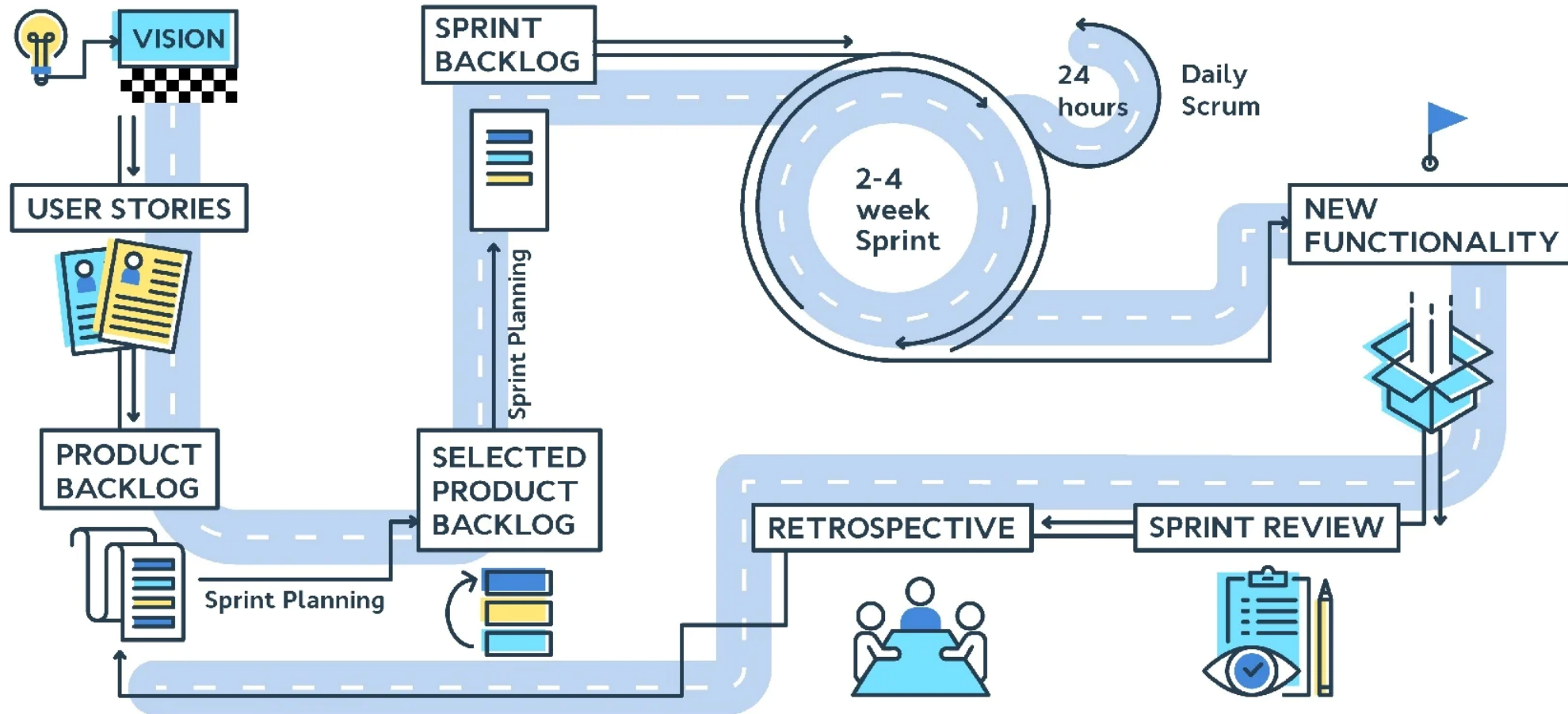
Echange et réflexion

Equipe

Sécurité affective

Conflance

SCRUM PROCESS



"Un récit utilisateur, ou « *user story* » en anglais, est une description simple d'un besoin ou d'une attente exprimée par un utilisateur et utilisée dans le domaine du développement de logiciels et de la conception de nouveaux produits pour déterminer les fonctionnalités à développer." *Wikipédia*

En tant que "rôle, persona, user"

Je veux "fonctionnalité, tâche, action"

Afin de "valeur ajoutée, résultat"

L'histoire est courte et écrite sur une carte

Les détails sont discutés par les équipes : ergonomes, développeurs, analystes, etc

L'histoire est confirmée par des tests d'acceptation rédigés au dos de la carte

EN TANT QUE CLIENT DU CA
JE VEUX RETIRER DU LIQUIDE D'UN DISTRIBUTEUR
AFIN DE DE NE PLUS FAIRE LA QUEUE À L'AGENCE

+ SCÉNARIO 1: LE COMPTE EST CRÉDITÉ

ÉTANT DONNÉ QUE LE COMPTE EST CRÉDITÉ
ET QUE LA CARTE EST VALIDE
ET QUE LE DISTRIBUTEUR CONTIENT DU LIQUIDE
LORSQUE LE CLIENT DEMANDE DU LIQUIDE
ALORS ON S'ASSURE QUE SON COMPTE EST DÉBITÉ
ET QUE LE LIQUIDE EST DISTRIBUÉ
ET QUE LA CARTE EST RENDUE

+ SCÉNARIO 2: LE COMPTE EST DÉBITEUR AU-DELÀ DE LA LIMITE DU DÉCOUVERT AUTORISÉ

Product Backlog

Il s'agit de la liste priorisée des besoins et des exigences du client : l'ensemble des fonctionnalités pour développer le produit

L'objectif est d'implémenter en premier les éléments qui ont le plus de valeur

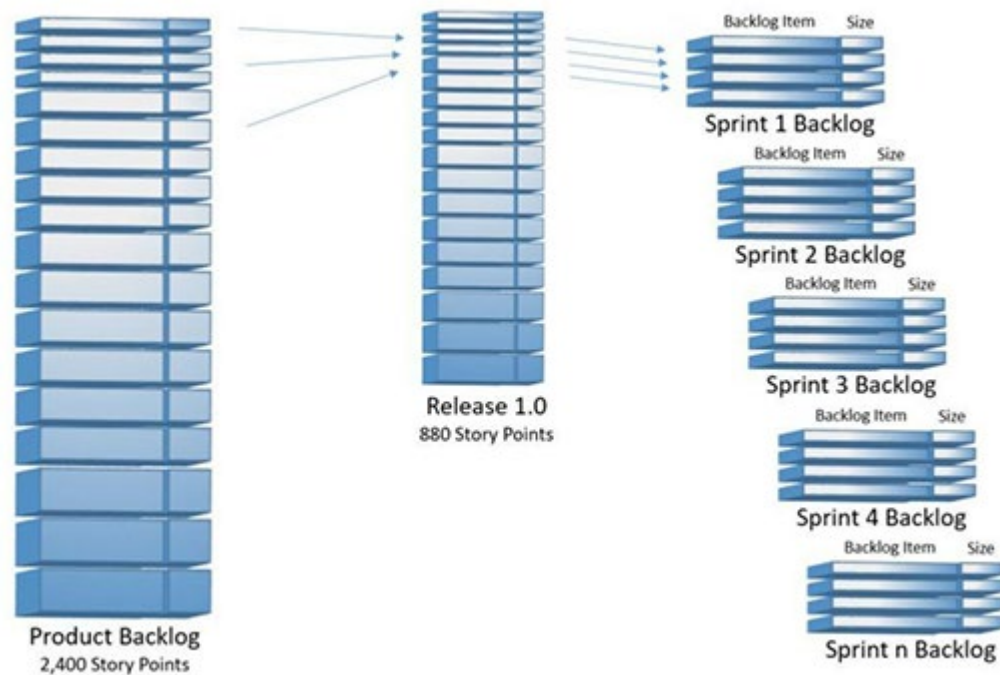
Mission essentielle du Product Owner

Exprime clairement les items du Product Backlog

Priorise les user stories pour mieux réaliser les objectifs et les missions

S'assure que le Product Backlog est visible, clair, compris, transparent pour l'équipe

Doliscrum					
			Edité par	JSI	Dernière mise à jour
					10/02/2020
	En vert : à compléter par le Product Owner / En jaune : à compléter par l'équipe				
Id	Titre	User story	Domaine	Priorité	Sprint
1	Authentication web page	As a user I need a start page to authenticate myself into the system	Security	100	1
2	Maven project	As a developer I want a default pom that compiles, package and start my application with a small web server	Productivity	90	1
3	User Story	As xx I want to do yyy because ZZZ and I will test this as follow : XXXXX			



- Définir les stories haut niveau (epics)
- **Ready** : définir les conditions nécessaires pour qu'un item du backlog soit considéré comme suffisamment décrit pour entrer dans un sprint (compréhension de tous)

Sprint

Itération : de 2 à 4 semaines

Durée constante

Enchainement

Un sprint contient :

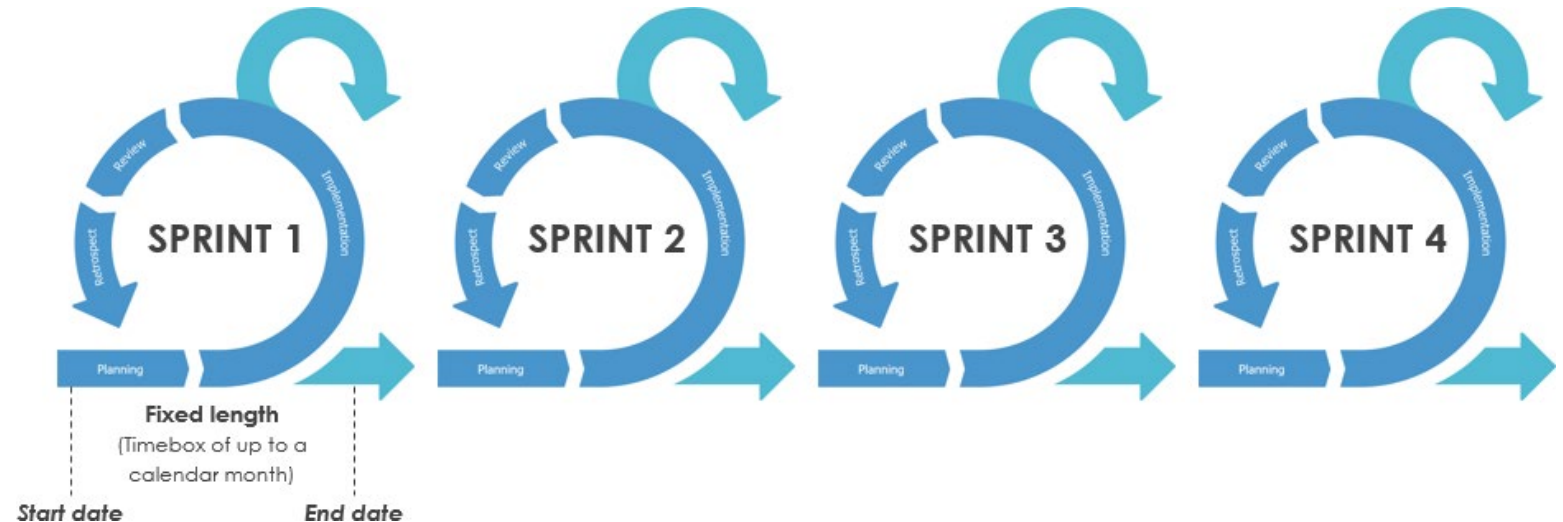
La planification du sprint (Sprint Planning)

Les mêlées quotidiennes (Daily Scrum)

Le développement (conception, tests, etc)

La review

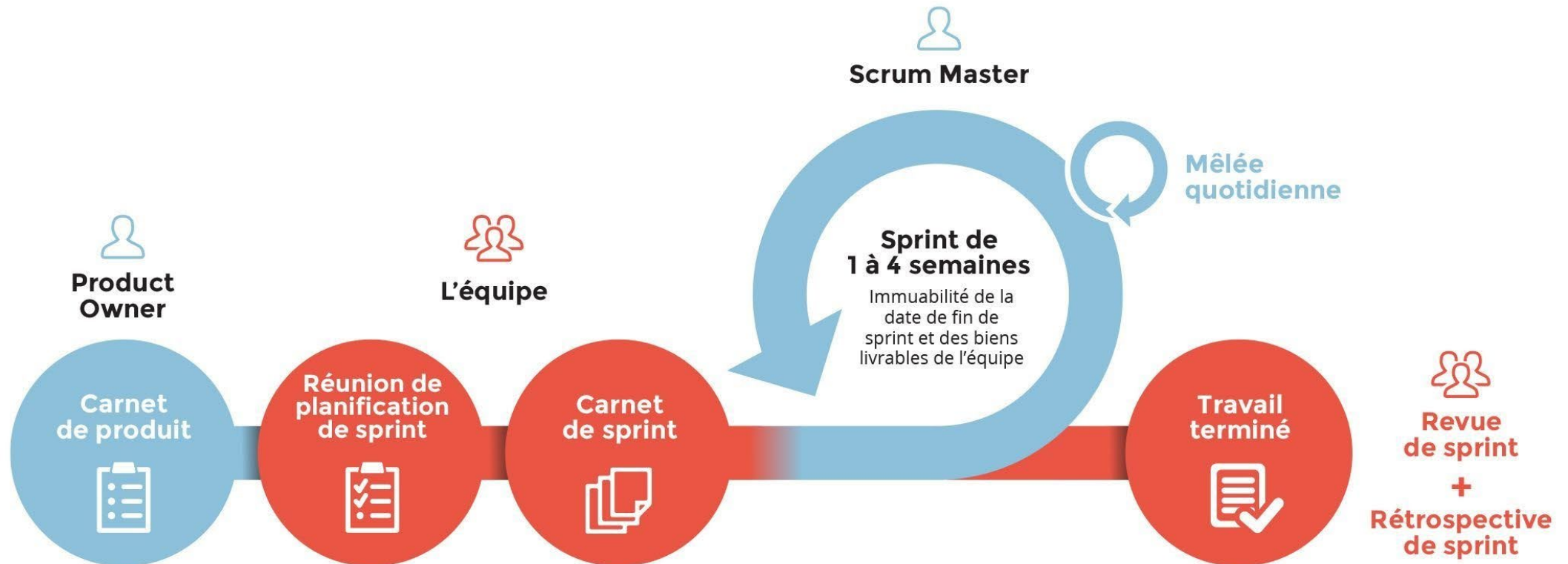
La rétrospective



Planification / Sprint Planning :

- Carnet de sprint / Sprint backlog (tâches priorisées)
- Planning Poker : donner une valeur à la tâche (difficulté)

Qu'est ce qui peut être terminé au cours de ce sprint ?
Comment sera effectué le travail choisi ?



Sprint Planning

En premier lieu, l'équipe se focalise sur la compréhension de ce que souhaite le PO et pourquoi il le souhaite

Le PO et l'équipe passent en revue les éléments de forte priorité du Product Backlog

L'équipe définit le but du Sprint

L'équipe estime la complexité et s'engage sur des user stories (validation)

Découpage des User en tâches dont le contenu et l'intérêt est compris par tous

Sprint Backlog

Rend visible le travail que l'équipe de développement doit réaliser pendant le sprint

L'équipe le modifie tout au long du sprint

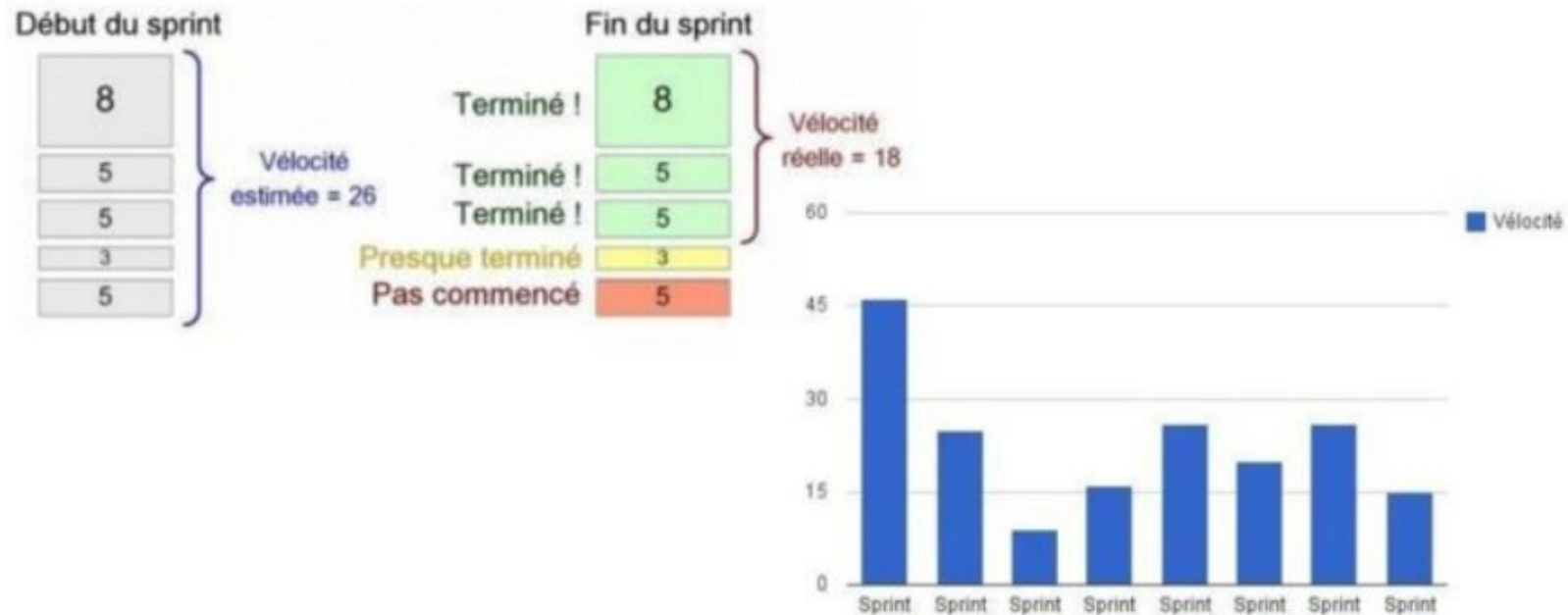
Le Scrum Master s'assure de la progression

Planning Poker

Estimations sur l'effort de développement des fonctionnalités.

La **vélocité** est un indicateur qui permet de déterminer l'effort qu'est capable de fournir une équipe de développement pour la réalisation des tâches programmées dans un sprint. Elle est exprimée en nombre de points.

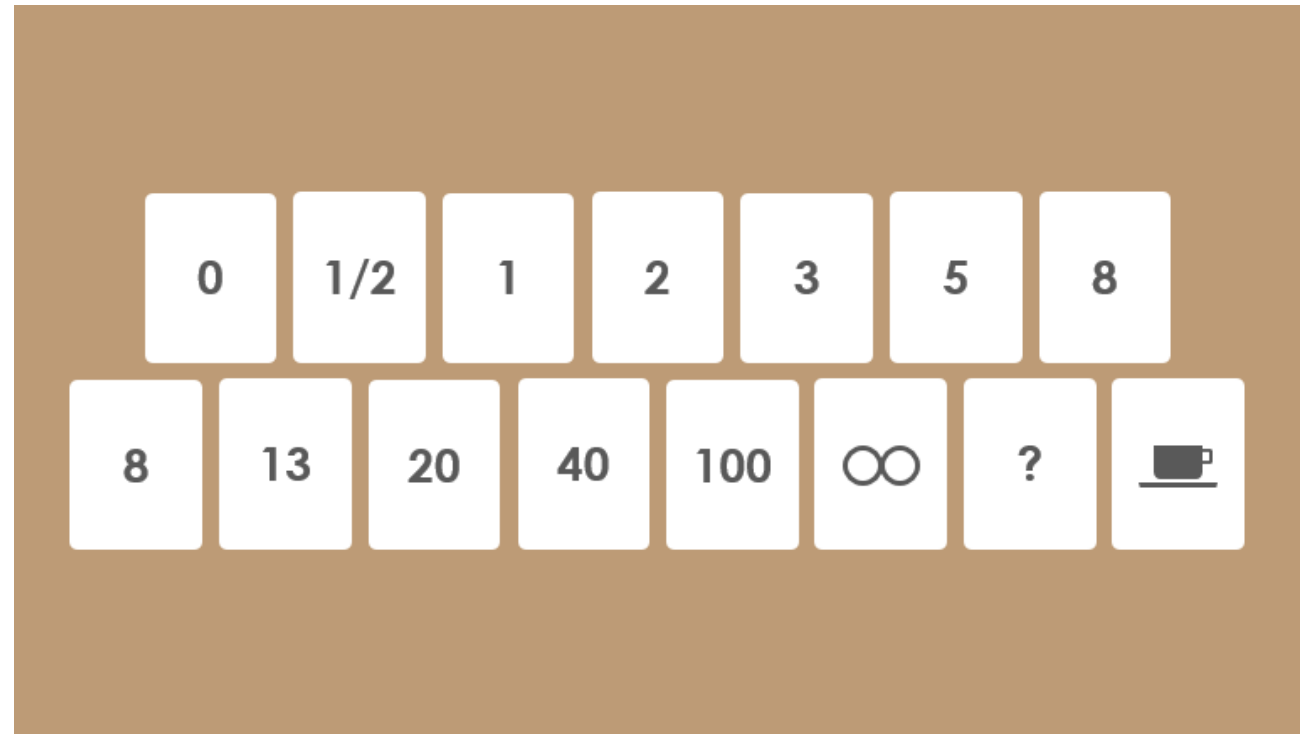
L'équipe de développement attribue à chaque items du product backlog item un certain nombre de points. Ces points représentent à la fois la complexité et la durée de la réalisation de la tâche, estimées de façon empirique.



La **suite de Fibonacci** est souvent utilisée.

Aussi, les valeurs pouvant être attribuées sont:

1 pour une tâche extrêmement simple, comme la correction d'un libellé par exemple,
2, 3, 5 pour une tâche légèrement plus complexe, comme la création d'un formulaire de saisie simple,
8, 13, 21, 34, 55, 89, 144 si l'on ne dispose pas de suffisamment d'informations pour estimer correctement cette tâche.



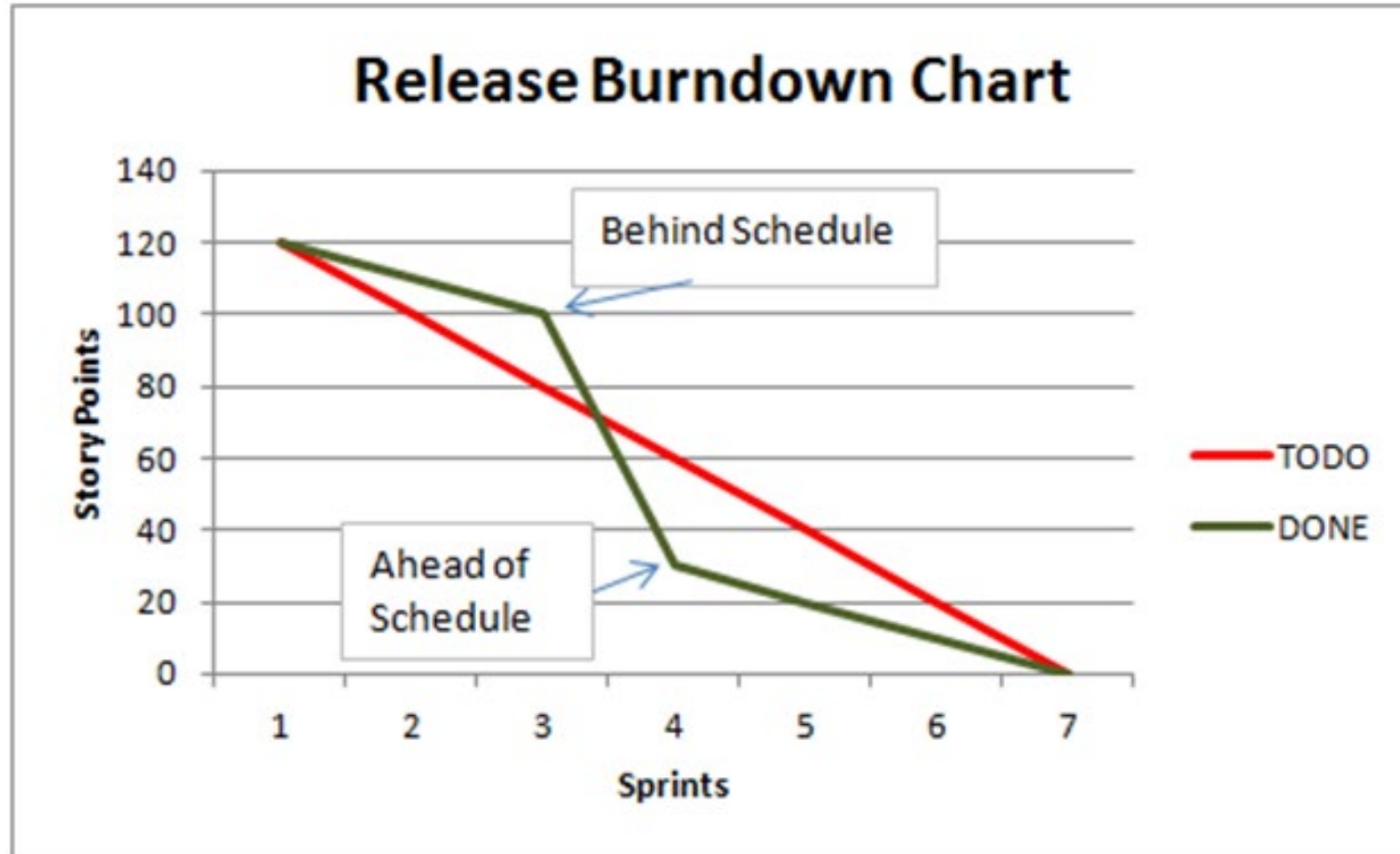
- Les participants s'installent autour d'une table
- Le responsable de produit explique à l'équipe un scénario utilisateur (*user story*)
- Les participants posent des questions au responsable de produit, discutent du périmètre du scénario, évoquent les conditions de satisfaction qui permettront de le considérer comme "terminé"
- Chacun des participants évalue l'effort de développement de ce scénario, choisit la carte qui correspond à son estimation et la dépose, face vers le bas, sur la table devant lui
- Au signal du facilitateur, les cartes sont retournées en même temps.
- S'il n'y a pas unanimité, la discussion reprend*.
- On répète le processus d'estimation jusqu'à l'obtention de l'unanimité.

*Expliquer les points de vue respectifs.

Ces explications achevées et comprises de tous, une nouvelle estimation est produite et c'est alors la moyenne arithmétique de ces estimations qui est prise en compte.



Burndown Chart



Pendant le sprint :

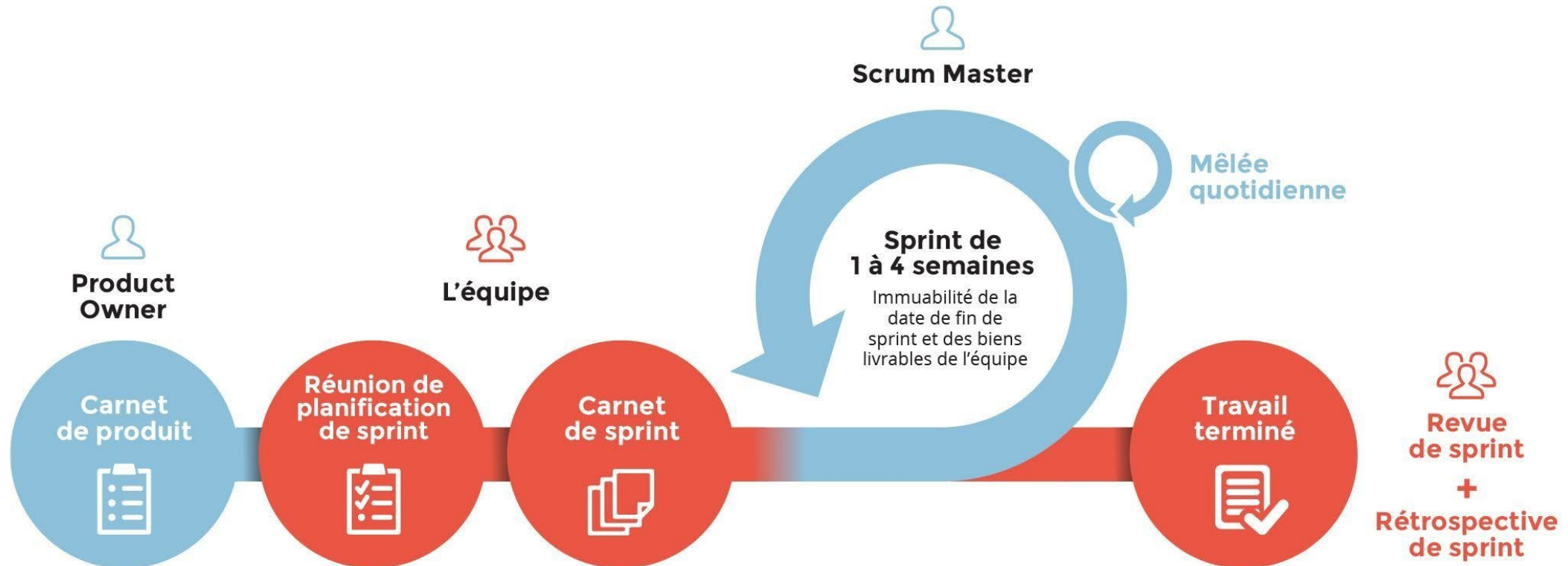
- La mêlée quotidienne / daily scrum : 15 minutes / 3 questions :

Ce que j'ai fait hier

Ce que je vais faire aujourd'hui

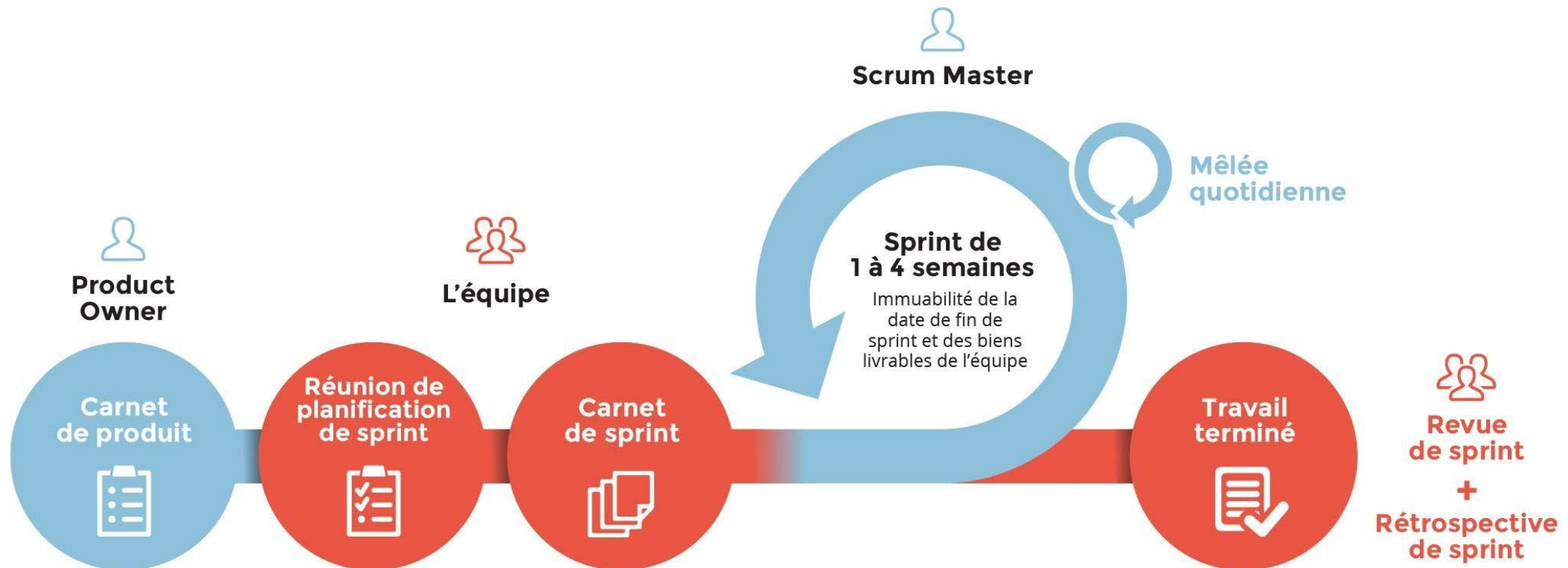
Si j'ai un problème/blocage

- Outils : Bundown **Chart** / **Kanban**



A la fin du sprint :

- Review : présentation au client (avec une démo)
- Rétrospective : en équipe (ce qui a fonctionné ou non, ce qu'il faut développer sur un prochain sprint, etc)



REVIEW

- Réunion organisée par le PO avec le client, présence de l'équipe (informelle)
- Inspecte l'incrément réalisé (démonstration)
- Réviser le Product Backlog si nécessaire
- Préparer le prochain sprint

RETROSPECTIVE

Inspection et adaptation

Amélioration ?

Fonctionnement ?

Équipe ? (radar de satisfaction pour le SM)

Comment régler les problèmes ?

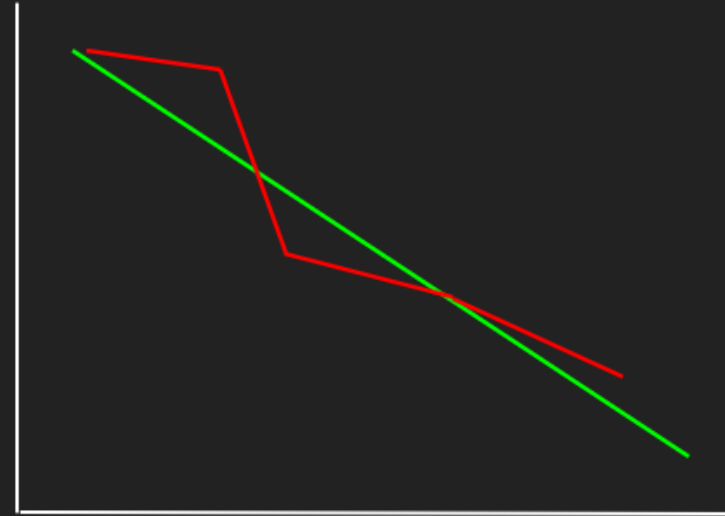
C'est le SM qui organise et facilite la réunion

Workflow

Burndown chart

MoodBoard

Kanban



SEPTEMBER 2011							
	1	2	3	4	5	6	7
BERND	😊	😐	🚧	🚧	😐	😊	😞
MARIKO	😊	😊	🚧	🚧	😞	😊	😊
ALEX	😊	😐	🚧	🚧	😊	😐	😞
MEIKE	😞	😊	🚧	🚧	😊	😊	😞
STEFAN	😊	😊	🚧	🚧	😐	😐	😊

Kanban

Mise au point dans les années 1970 chez Toyota

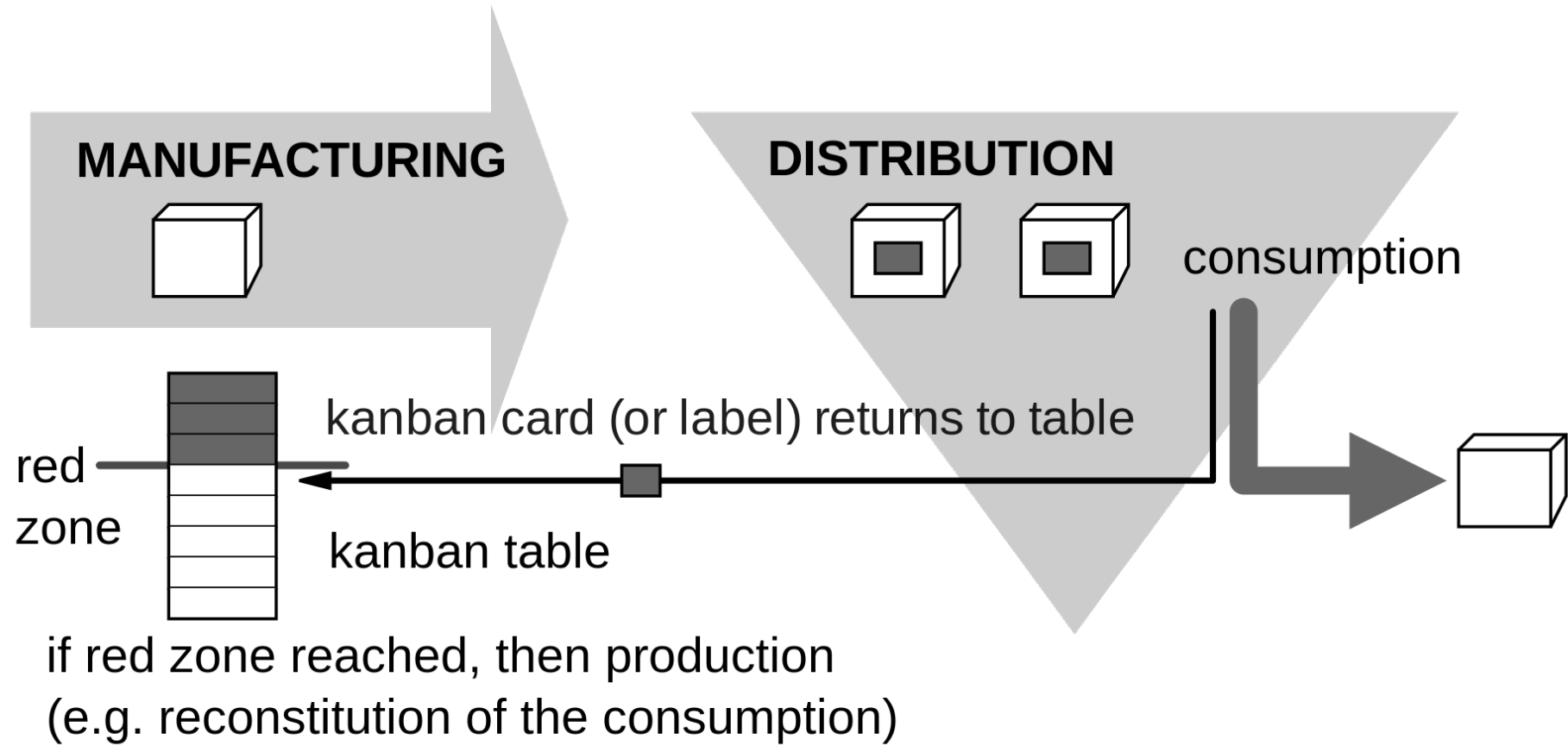
Mot japonais traduisible par « enseigne » et souvent traduit, pour la méthode Kanban, par « étiquette »

Comment produire à bon escient et juste en fonction des besoins ?

Cette méthodologie prévoit un système de production basé sur la demande plutôt que sur l'offre. Le client commande, le fournisseur produit pour répondre à sa demande.

Avantages :

- Éviter la surproduction et donc les invendus.
- Limiter les coûts de stockage.
- Être capable d'améliorer les processus au fur et à mesure.
- Pouvoir rapidement corriger un défaut sur un produit ou résoudre un problème dans la chaîne de production.



User Stories



To do



In Progress



Done



Ready



Sprint

Done



Product increment

Tâche 1



4 principes du Kanban

Commencer par ce que vous faites actuellement

Appliquer des changements évolutifs et augmentés

Respecter les rôles et les processus en place

Encourager le leadership



Visualiser le travail / Transparence

3 étapes classiques : Tâches, In progress, Done + autres étapes dans vos processus actuels (test, blocage)

Limitation des tâches en cours

Il ne faut pas noyer le board de tâches mais le limiter quotidiennement de manière raisonnable

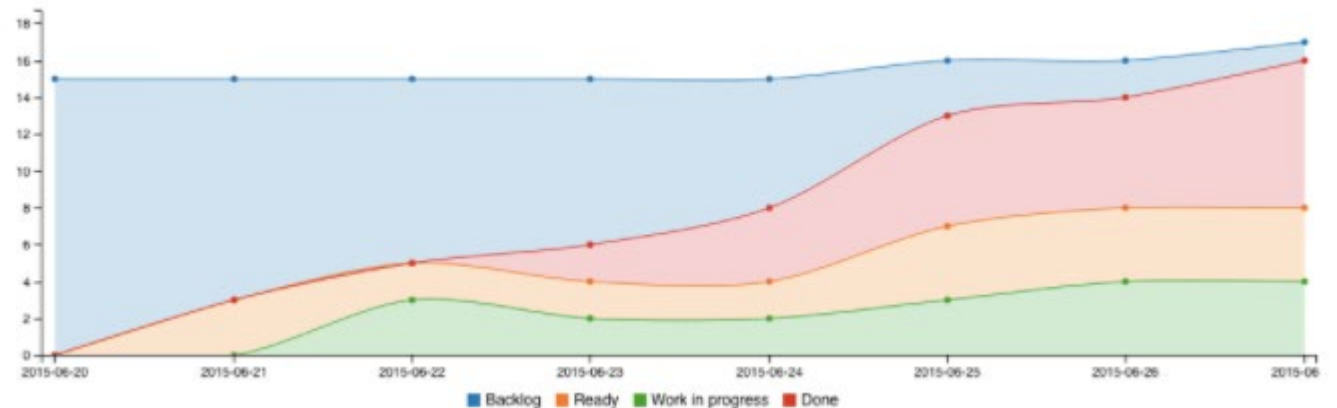
Gestion du flux

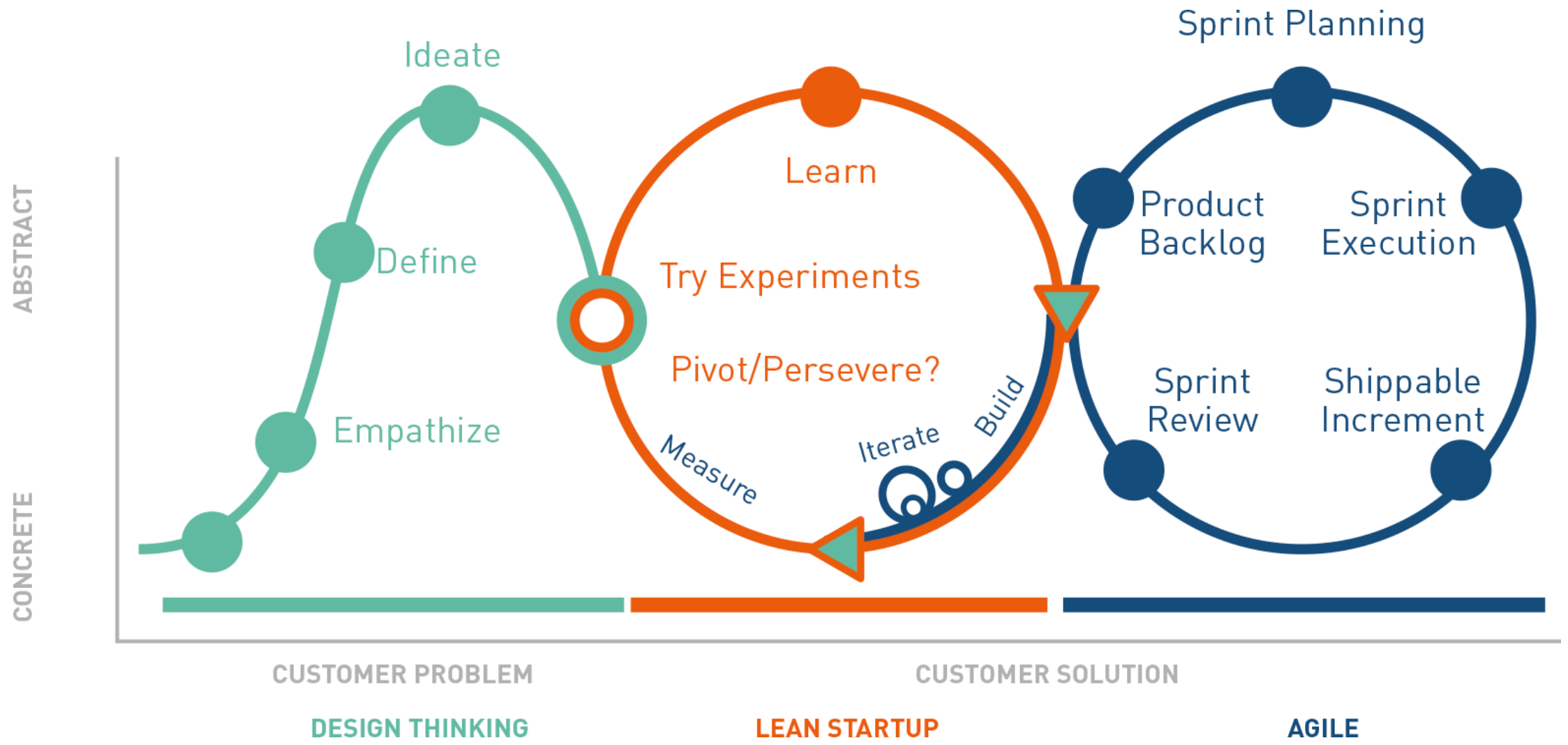
Mesurer l'avancement des taches pour chaque état différent.

Règles de déplacement

Déterminer les critères de changement d'un état à un autre.

Utiliser des modèles pour reconnaître les opportunités d'amélioration





Cahier des Charges

Modalités : en équipe (4 groupes)



Livrables :

- Une présentation du déroulé de votre projet
- Un prototype d'application web répondant au cahier des charges
- Pitch de votre projet en fin de sprint

Objectif :

- Travailler en équipe en appliquant les notions abordées aujourd'hui
- Utiliser les outils de gestion de projet (Workflow : Kanban, Lean Canvas)

Constitution de votre équipe

Attribution des rôles (1 PO / 1 SM / Equipe de dev)

Analyse du cahier des charges

Mise en place de vos outils : Kanban

Réalisations : Lean Canva, Persona, User Stories...