



SCRUM :

le cadre
méthodologique de
référence

Mame Diarra Mbaye
Référente-Digitale

PLAN



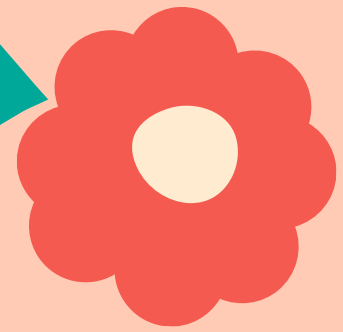
histoire, définitions,
contextes et enjeux

Initialisation d'un projet
Agile

Scrum : Bases et rôles

Scrum : Définir le besoin





CONCEPTS :



histoire, définitions,
contextes et enjeux

Approches prédictives
vs Approches
empiriques

DE WALT DISNEY

"





Quand nous entamons un nouveau projet, nous y croyons jusqu'au bout. Nous avons confiance en la capacité de le réussir.

"

DÉFINITION D'UN PROJET

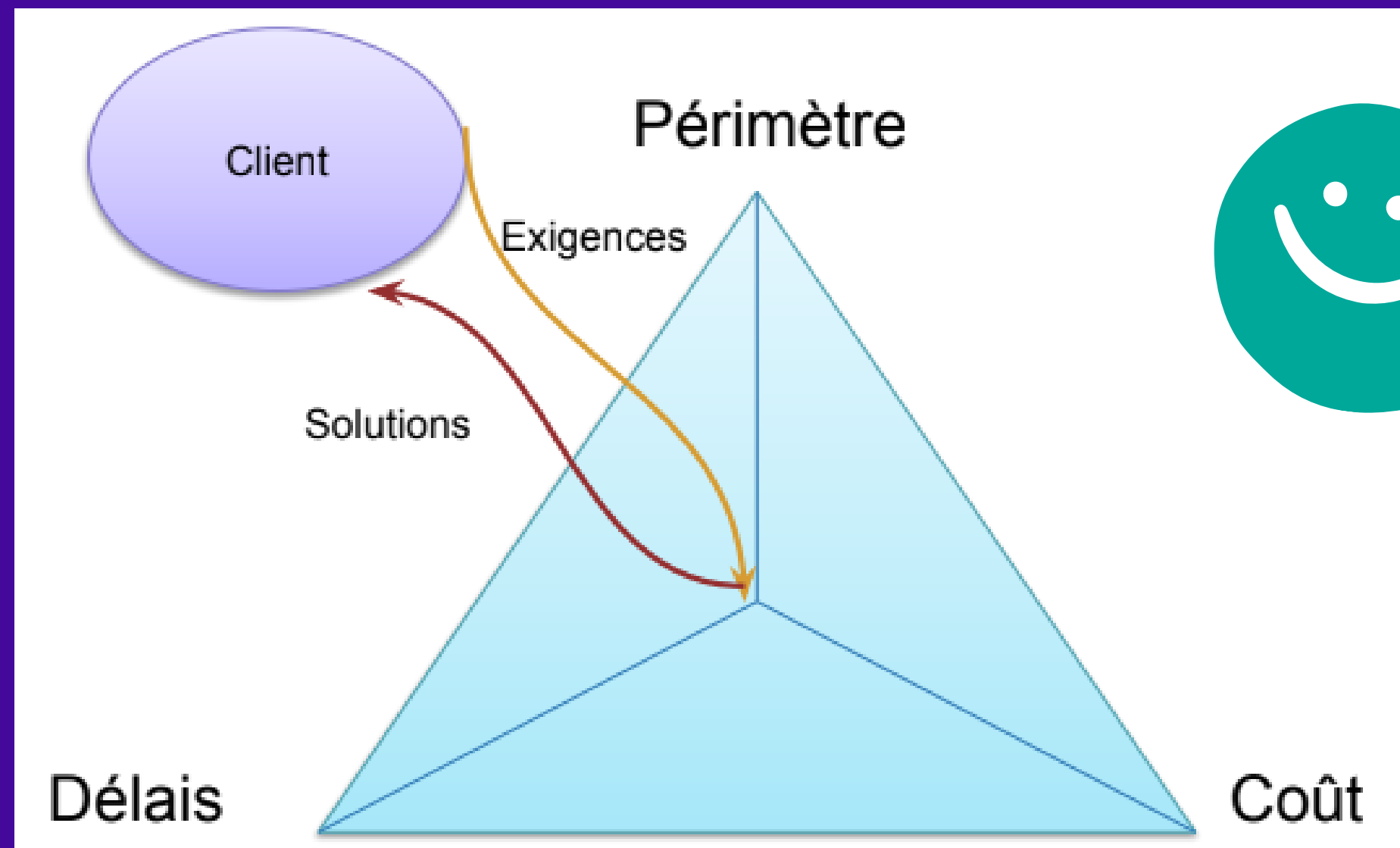
Rappel

Un projet est un objectif **temporaire** décidé en vue de produire un résultat **unique**, avec des ressources limitées.

-  Objectif : le projet a une cible définie et mesurable.
-  Temporaire : Tout projet a une date de début et de fin
-  Unique : Le produit ou le service possède des traits distinctifs de tout autre produit ou service similaire
-  Ressource limitée : Le projet s'exécute avec des ressources (temps, hommes, argent) finies.

CONTEXTE ET ENJEUX

le triangle de qualité



APPROCHE PRÉDICTIVE

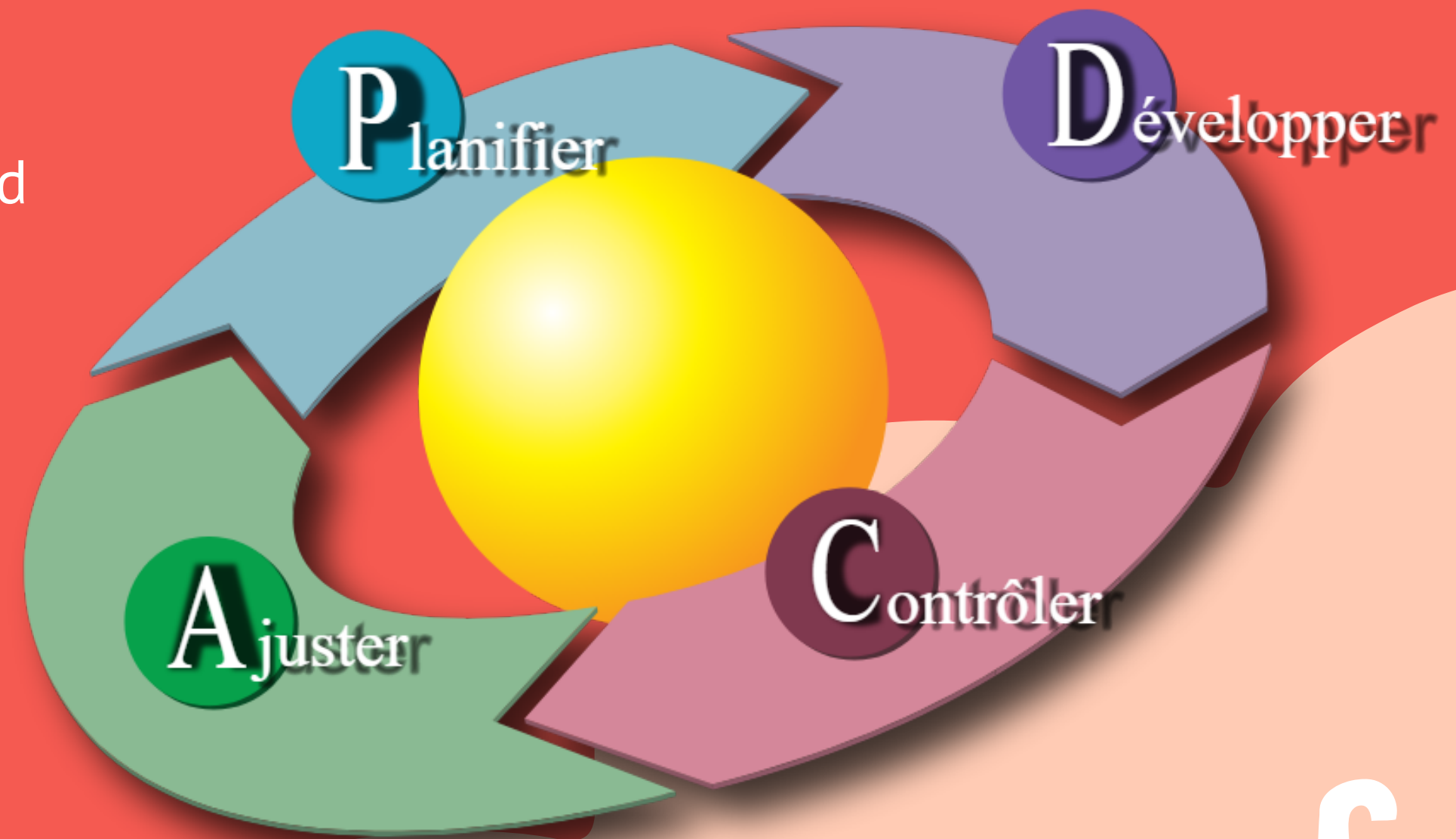


- ✿ Taylorisme (19eme siècle) : approche scientifique, pour la production de masse (pas d'imprévu). « Un bon ouvrier fait ce qu'on lui dit de faire et ne discute pas ».
- ✿ Fordisme (Début 20eme siècle) : Standardisation de la production de masse.



APPROCHE EMPÉRIQUE

- ❁ Deming (Années 50) : Amélioration continue, abandon du « command and control ».
- ❁ Toyota (Années 70) : Amélioration managériale (Kaizen et Lean), visualisation des tâches de l'équipe (Kanban).



PRÉDICTIF VS EMPÉRIQUE

llll

☐ « Plan the Work,
Work the Plan »

☐ « Fail Safe, Fail
Fast »

❖ Tout est prévu à l'avance.

❖ Basé sur l'expérience.

La différence ?
Le droit à l'erreur !

QUELLES APPROCHES DE GESTION DE PROJET PRATIQUEZ OU CONNAISSEZ- VOUS ?



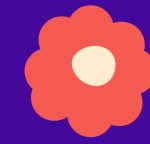
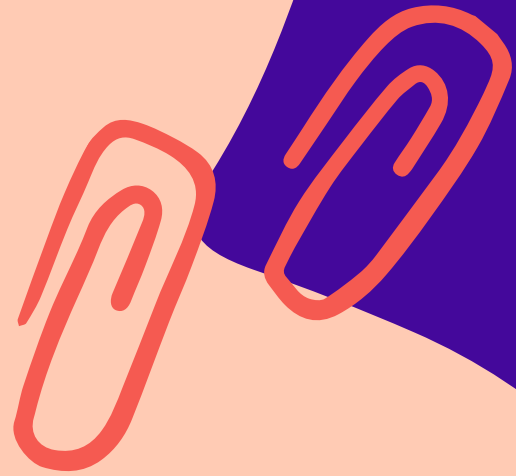
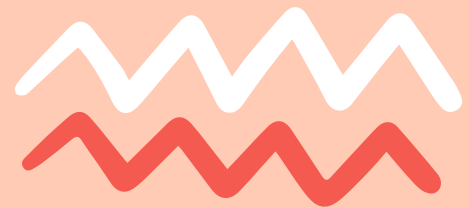
CYCLE EN CASCADE



Visez l'excellence
dans le
développement

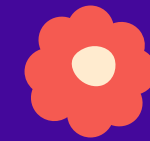


Cycle en Cascade



Avantages :

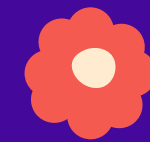
Très séquentiel et simple à comprendre.



Désavantages :

Méthodologie rigide, donc inutilisable en cas de nombreux changements.

Peu efficace (d'après le Standish Chaos Report) et de moins en moins utilisé.



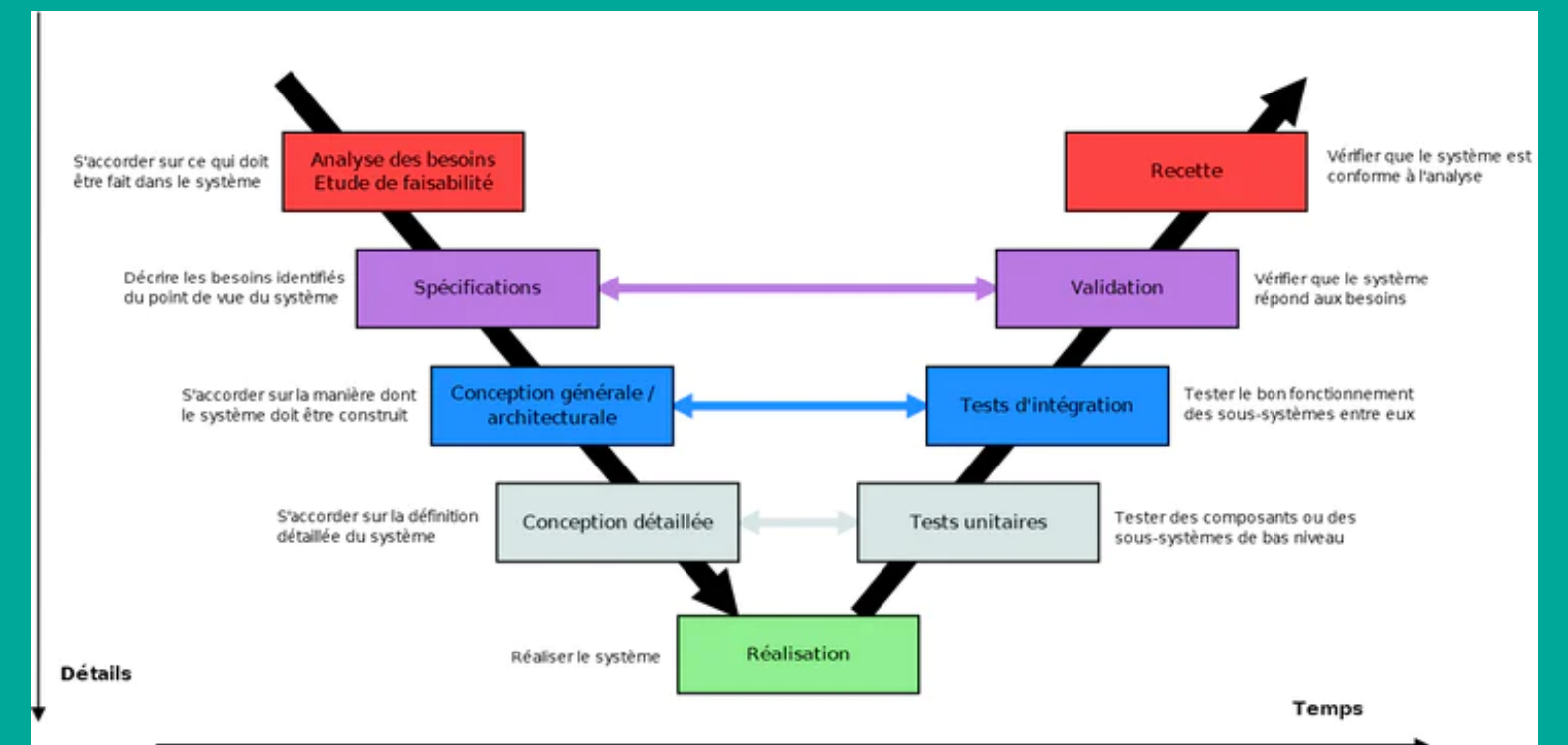
Conclusion : Peu efficace dans des projets complexes.

Utilisé principalement dans le BTP.

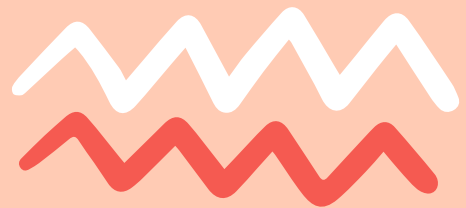
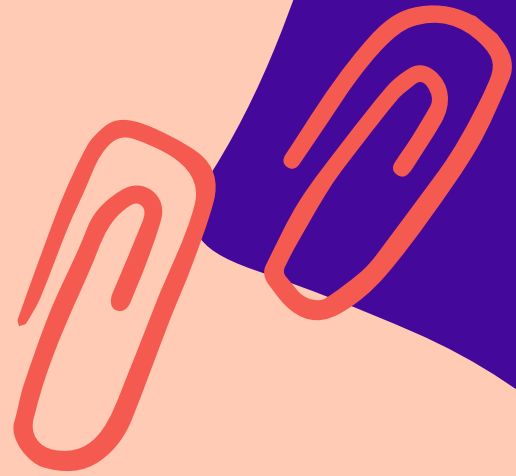
CYCLE EN V



Visez l'excellence
dans le
développement



Cycle en V



Avantages :

Assez séquentiel et rigoureux.
Attitude proactive dès le départ du projet.
Très utilisé dans l'armement et reconnu.



Désavantages :

Méthodologie rigide, inefficace en cas de changements.
Documentation très lourde.
Organisation lourde : architectes, testeurs, MOA déléguée, assistance à MOA, MOA technique, direction projet...

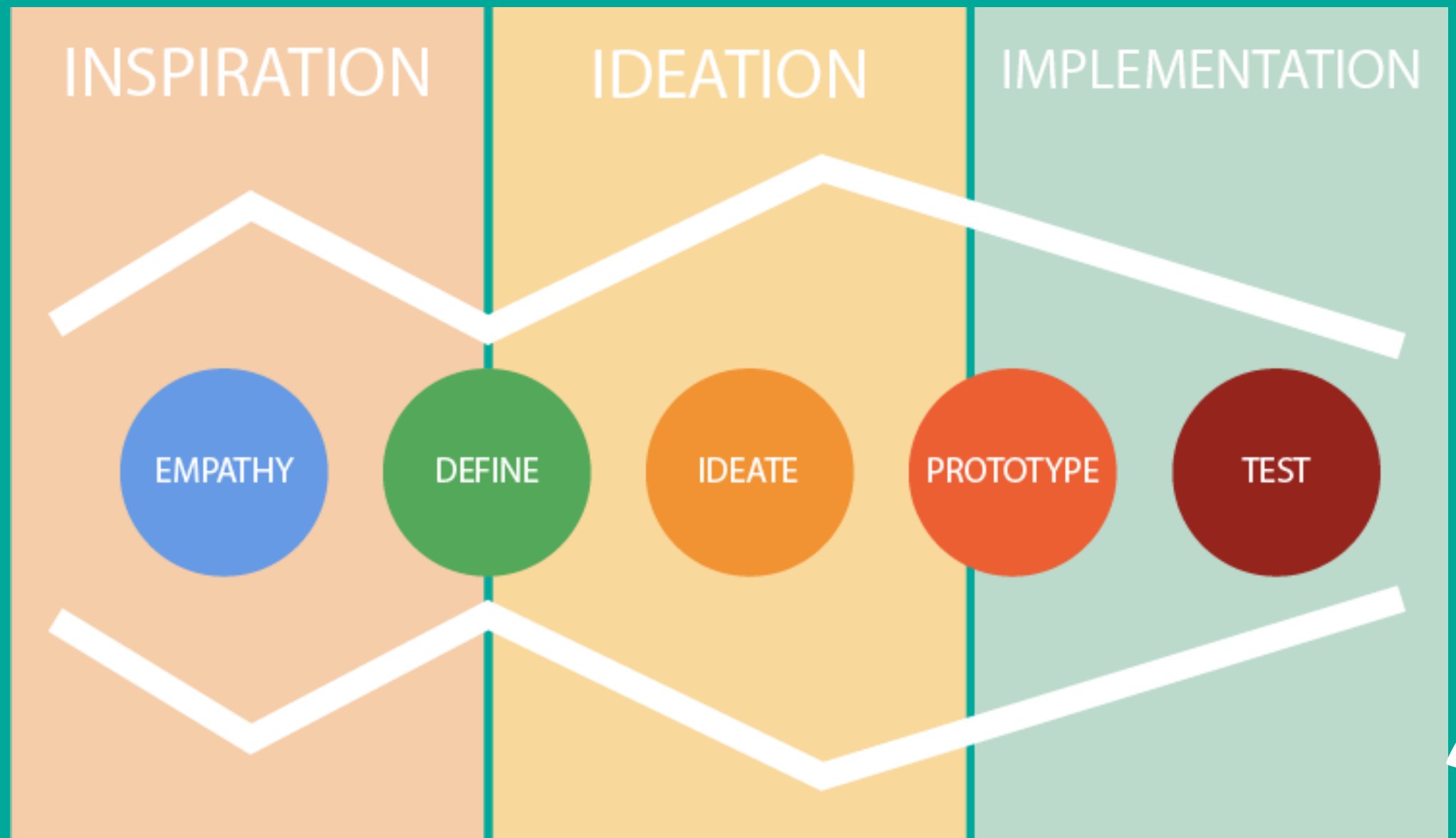


Conclusion : utilisable dans un contexte peu changeant
(ex : armement, aéronautique).

DESIGN THINKING



Visez l'excellence
dans le
développement





Design Thinking

- ✿ Avantages :
Axé client via des interviews.
Processus très créatif.
- ✿ Désavantages :
Nouvelle façon de gérer des projets, et donc peut provoquer une rupture complexe. Moins rigoureux que Kanban / Scrum.
Peu d'adeptes par rapport aux méthodes agiles classiques.
- ✿ Conclusion : puissant dans un contexte inconnu (innovation) et dans une culture de développement personnel.

LES 4 VALEURS D'AGILITÉ

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

[agilemanifesto.org principles.html](http://agilemanifesto.org/principles.html)

▀ 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.

TYPES DE MÉTHODOLOGIES AGILE,

ET LES POINTS INTÉRESSANTS:

LEAN



Penser
amélioration
continue et qualité,
réduire voire
supprimer les
étapes avec peu de
plus-value

SCRUM



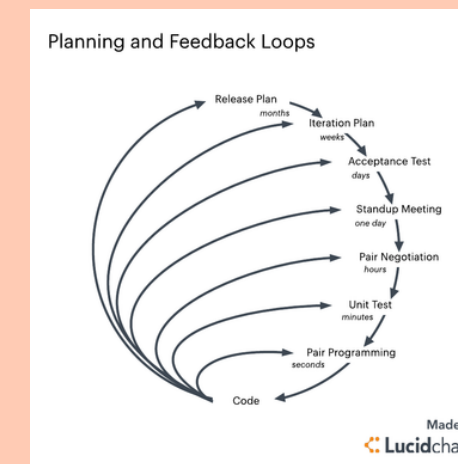
Travail en itération,
découper les
tâches,
auto-organisation,
« Stand-up »
quotidien

KANBAN



Visualiser les
tâches de toute l'
équipe dans un
processus
identifié, et gérer
le surplus de
tâches

XP

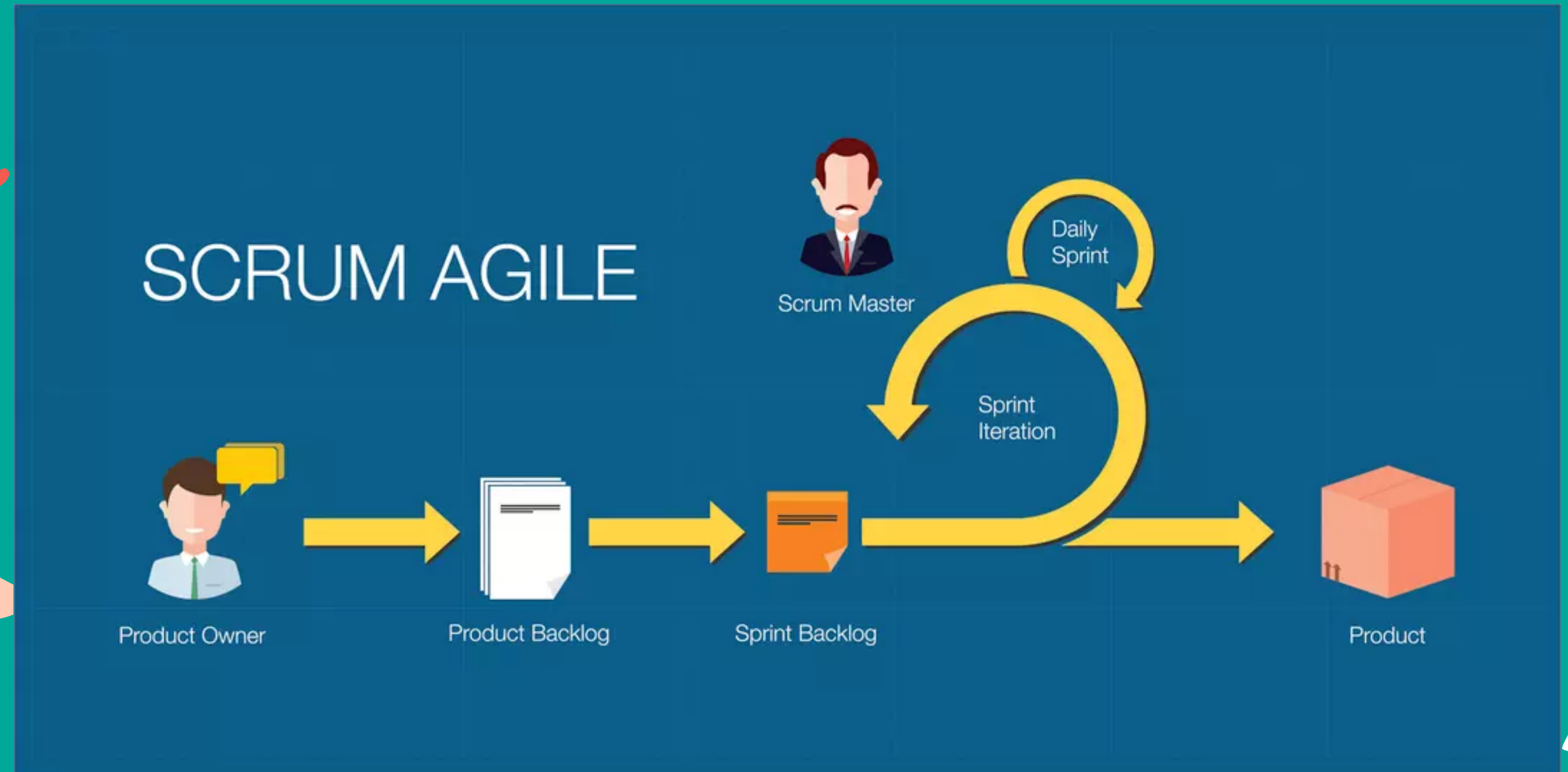


Réaliser
régulièrement des
tests unitaires,
avoir un code
« clean » et
travailler par paires

CYCLE CRÉATIF AGILE



SCRUM



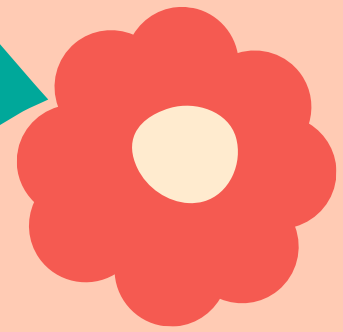


Cycle créatif agile

SCRUM



- 🌸 Avantages :
 - Accueille le changement au cours d'un projet.
 - Equipes autoorganisées = motivées.
 - Amélioration continue par des rétrospectives.
 - Prône une relation continue avec le client et les sponsors via des démos.
- 🌸 Désavantages :
 - Difficile de donner un budget prévisionnel.
 - Complexe à mettre en œuvre dans une culture de contrôle (plus de chef de projet !).
- 🌸 Conclusion : efficace dans un contexte changeant (ex : dév informatique)



CONCEPTS :



Initialisation d'un projet
Agile

S'orienter utilisateur
et bien commencer



Initialisation : Interview client



- Interview avec le client et le sponsor.
- Objectifs :
 - S'accorder avec le client puis le sponsor sur les objectifs et contexte.
 - Connaitre l'organisation, les futurs utilisateurs, le but du produit.
 - Identifier les critères de choix (prix, organisation d'équipe, etc...).
 - Identifier les contraintes imposées.
- Outil utilisé : le QQCOQP.



Initialisation : le QQCOQP

■ Se poser les bonnes questions :

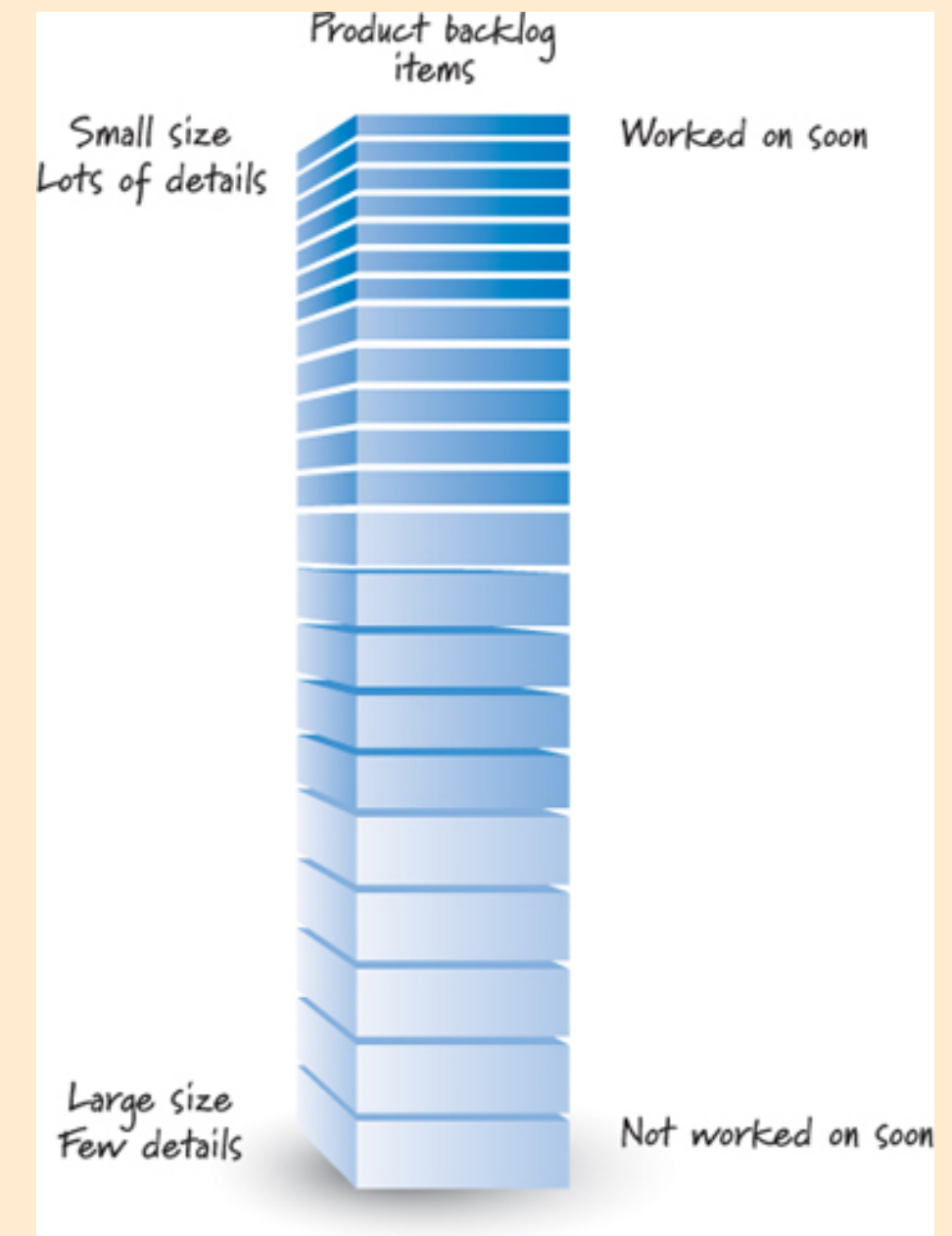
- ☐ Qui : Qui est le client du projet ? Qui seront les utilisateurs ?
- ☐ Quoi : Est-ce un projet nouveau ? Quelles sont les données d'entrées ?
- ☐ Comment : Quels sont les moyens (financiers / ressources / techniques) ?
- ☐ Où : Où sera effectué le projet ? Sur quelle technologie ?
- ☐ Quand : Quand doit démarrer le projet ? Se terminer ?
- ☐ Pourquoi : Pourquoi le projet est-il lancé ?

Initialisation : et en Agile... ?

Un cahier des charges agile correspond simplement au besoin des utilisateurs et à la vision du produit, au jour le jour.

■ Il faut donc pouvoir gérer le changement.

■ En Agile, c'est le Product Backlog.



Le 1er Sprint

- En Agile, on fonctionne en Itérations : des Sprints.
- Il faut commencer à délimiter les rôles :
 - ☐ Le Product Owner : Fonctionnel
 - ☐ Le Scrum Master : Organisationnel
 - ☐ L'équipe de développement : Technique



Le 1er Sprint

■ On définit deux choses importantes :

- ☐ La durée des sprints (2 à 4 semaines), choisie principalement en fonction des contraintes business et techniques.
- ☐ La Définition de « Fini » : la checklist pour définir lorsqu'une « user story » (tâche fonctionnelle) est terminée.



SCRUM :



Bases et rôles

Comprendre les méthodes
agiles pour
tendre vers l'auto-
organisation

DE WINSTON CHURCHILL.

"

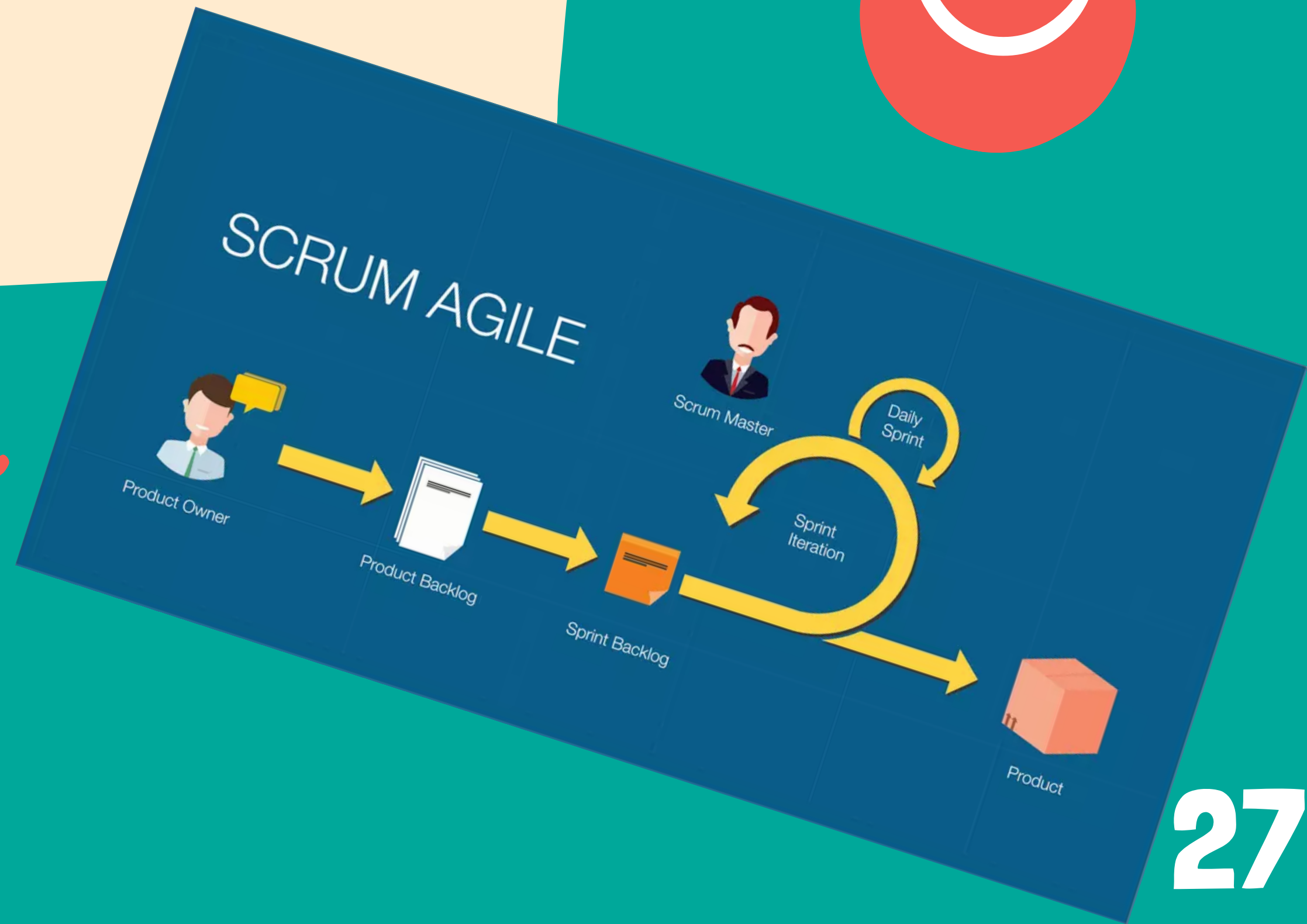
**Les plans n'ont que peu
d'importance, mais planifier est
essentiel**

"

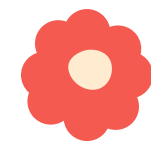
PARLON AGILE, PARLONS SCRUM !



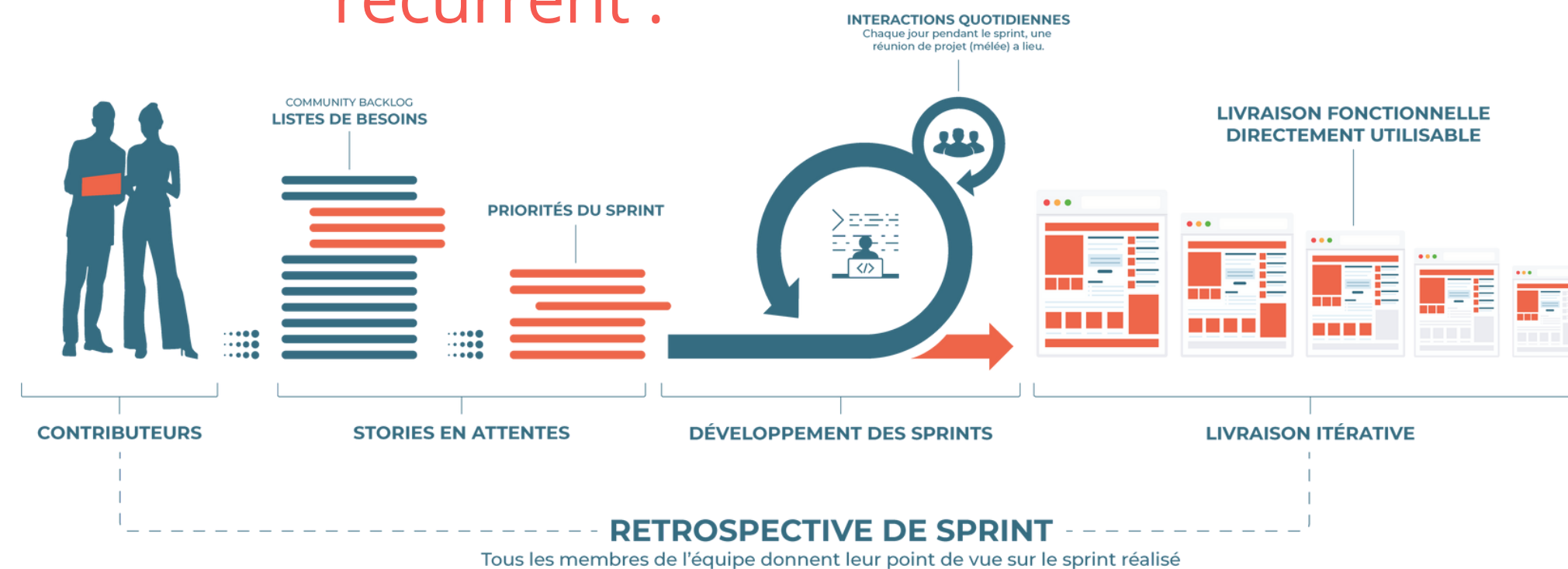
la méthode la plus pratique!



Parlons agile, parlons scrum !



Scrum est une méthode de management de projet créée afin de répondre au manifeste agile. Néanmoins, Scrum est de plus en plus utilisé dans d'autres domaines, comme le Marketing. L'idée est d'éviter ce phénomène récurrent :



Apprendre Scrum

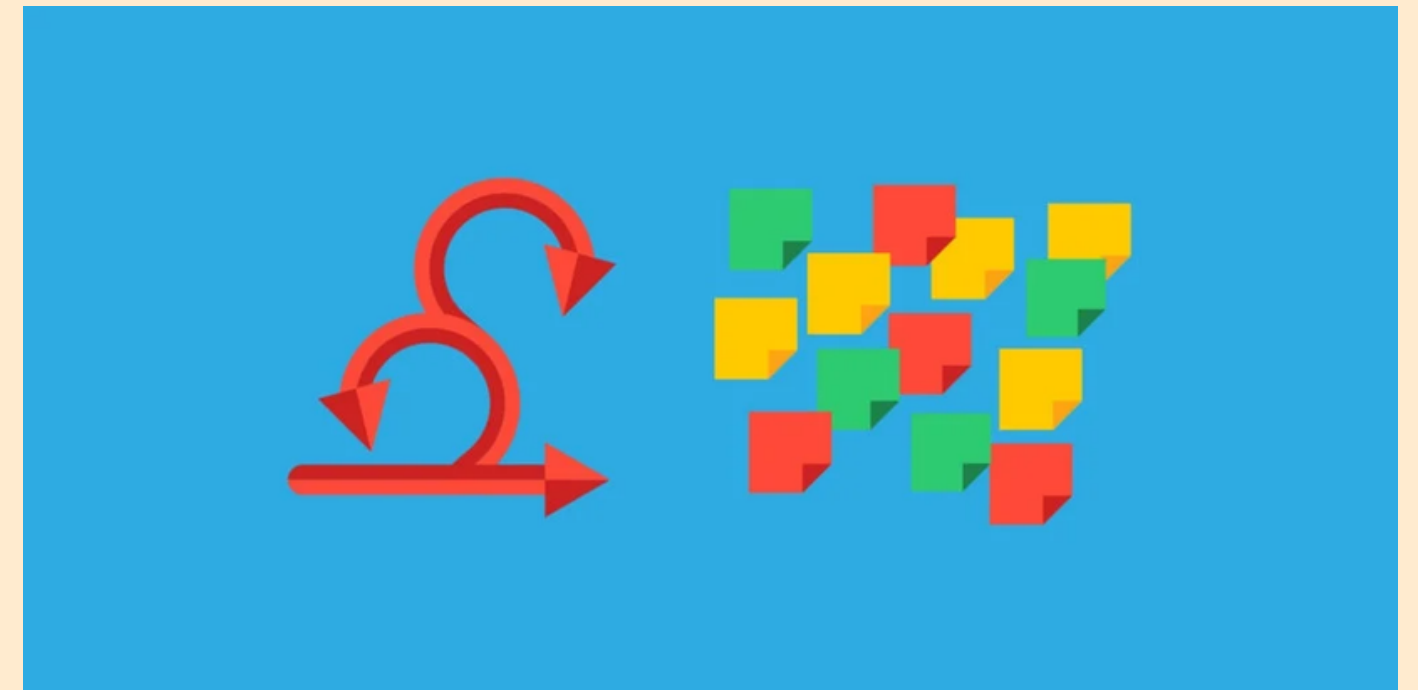
■ Scrum n'est pas une recette, ni un coup de baguette magique !

■ Scrum est un cadre de développement de produit :

- ❖ Léger,
- ❖ Simple à comprendre,
- ❖ Difficile à maîtriser.

■ Scrum est un cadre habilitant :

- ❖ Révèle les problèmes et les difficultés,
- ❖ Aide à les affronter, et à les solutionner...
- ❖ ...grâce à vos apports et à votre expertise.



Ce que contient Scrum :

3 rôles clés :



- ✓ Le Product Owner
- ✓ L'Equipe
- ✓ Le Scrum Master

5 activités



- ✓ Le Sprint
- ✓ Le Sprint Planning
- ✓ Le Daily Standup
- ✓ La Revue de Sprint
- ✓ La Rétrospective

5 artefacts :

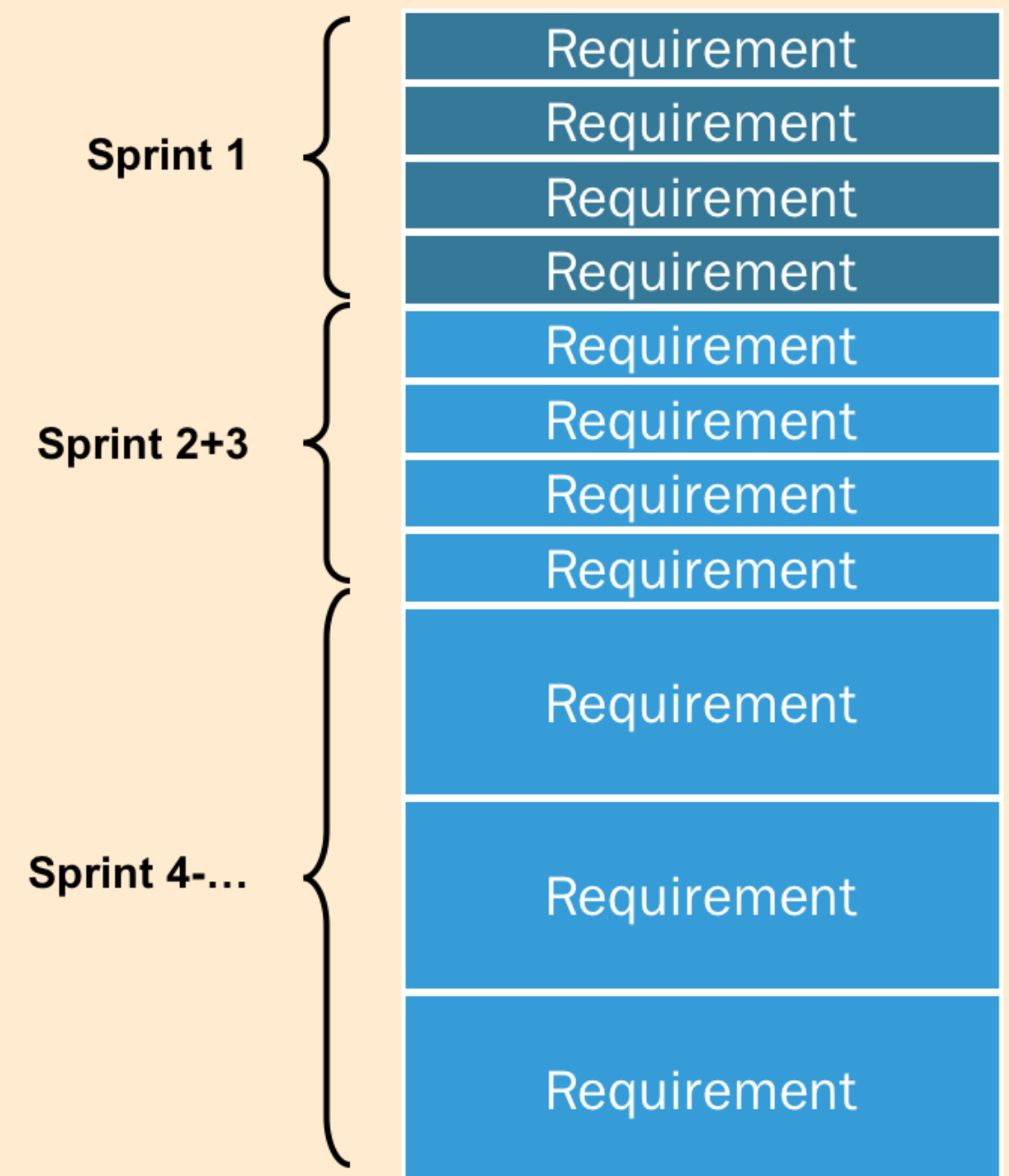
- ✓ Le Product Backlog
- ✓ L'indicateur d'avancement de produit et de sprint
- ✓ Le Sprint Backlog
- ✓ Le produit de l'itération
- ✓ La définition de Fini

L'équipe Scrum : le Product Owner (PO)

■ Le Product Owner est le responsable du produit, il va maximiser sa valeur et donc le travail de l'équipe de développement. Il a la meilleure vision de son produit.

■ Il est la seule personne capable de modifier le Product Backlog !

■ C'est sa responsabilité : il est la partie Fonctionnelle de l'équipe !



L'équipe Scrum : le Product Owner (PO)

■ Ses tâches sont multiples :

❖ Exprimer clairement à l'équipe, par des phrases simples, les besoins des utilisateurs : « En tant qu'utilisateur, je voudrais [...] pour [...] ». On appelle cela des

User Stories, qui pourront être ensuite traduites en tâches par l'équipe de développement

❖ Prioriser les US pour avoir un logiciel opérationnel rapidement,

❖ Avoir un Product Backlog visible et transparent pour tous,

❖ Prendre en considération les besoins des utilisateurs, des clients, des sponsors...

L'équipe Scrum : le Product Owner (PO)

■ C'est un rôle-clé dans la démarche Scrum !

■ Compétences souhaitées d'un bon Product Owner :

- ☐ Bonne connaissance des métiers de l'équipe
- ☐ La maîtrise des techniques de définition de produit,
- ☐ Capacité à partager sa vision du produit,
- ☐ Capacité à prendre des décisions rapidement et à effectuer des choix structurants,
- ☐ Capacité à détailler facilement les besoins utilisateurs,
- ☐ Un esprit ouvert au changement,
- ☐ Une aptitude à la négociation.



L'équipe Scrum : Les développeurs

■ Elle est constituée par tous ceux qui interviennent sur le produit :

❖ Développeurs, Architectes, Testeurs, Intégrateurs, etc.

■ L'équipe est la responsable du « Comment ? ». Ses responsabilités et activités :

- ☐ Elle s'engage sur la réalisation du produit lors d'un sprint,
- ☐ Elle estime l'effort nécessaire,
- ☐ Elle découpe en tâches les fonctionnalités demandées,
- ☐ Elle s'auto-organise en interne,
- ☐ Elle respecte les priorités fixées par le Product Owner,
- ☐ Elle collabore avec le Product Owner,
- ☐ Elle présente le résultat au Product Owner et aux utilisateurs.



L'équipe Scrum : Les développeurs

- Chaque membre se met au service de l'équipe, Aucune mise en valeur individuelle possible.
- A l'image du sport collectif : c'est toute l'équipe qui gagne ou qui perd !
 - Généralement composée de 3 à 9 membres
- Moins c'est bien, mais l'équipe doit être capable de devenir autonome. Elle doit donc avoir suffisamment de compétences pour réaliser le produit !
- Hétérogène, mais tend vers l'équilibre.
- Autonome, s'organise comme elle l'entend.
- La plus stable possible.

L'équipe Scrum : le Scrum Master

■ Il a le rôle de facilitateur : 1 Scrum Master par équipe Scrum.

■ Ses responsabilités et activités :

- ☐ Il est impliqué sur l'organisation de l'équipe,
- ☐ Il est engagé sur la progression de l'équipe,
- ☐ Il veille à la bonne marche de Scrum, et anime l'équipe,
- ☐ Il élimine les obstacles qui peuvent réduire l'efficacité de l'équipe, et la protège de toute interférence externe,
- ☐ Il s'assure que chacun respecte son rôle,
- ☐ Il s'assure de la collaboration entre les intervenants,
- ☐ Il veille à ce que le Product Owner soit toujours en mesure d'alimenter l'équipe, notamment avant chaque Sprint planning !
- ☐ Il veille à ce que l'équipe se pose les vraies questions et affronte les problèmes qu'elle rencontre lors de rétrospectives



L'équipe Scrum : le Scrum Master

- ☐ Le Scrum Master n'est pas un chef de projet !
 - ▷ L'équipe est en auto-gestion : elle choisit elle-même ses tâches.
 - ▷ Il n'a pas d'autorité ni sur l'équipe, ni sur le PO. Il forme l'équipe à Scrum et à ses rituels, et explique pourquoi les choses se font ainsi.
- ☐ Il n'impose pas, il fait confiance.
 - ▷ Il peut être force de proposition, mais ne doit pas imposer de décision.
- ☐ Il ne contrôle pas, il facilite.
- ☐ Il ne dirige pas, il accompagne.

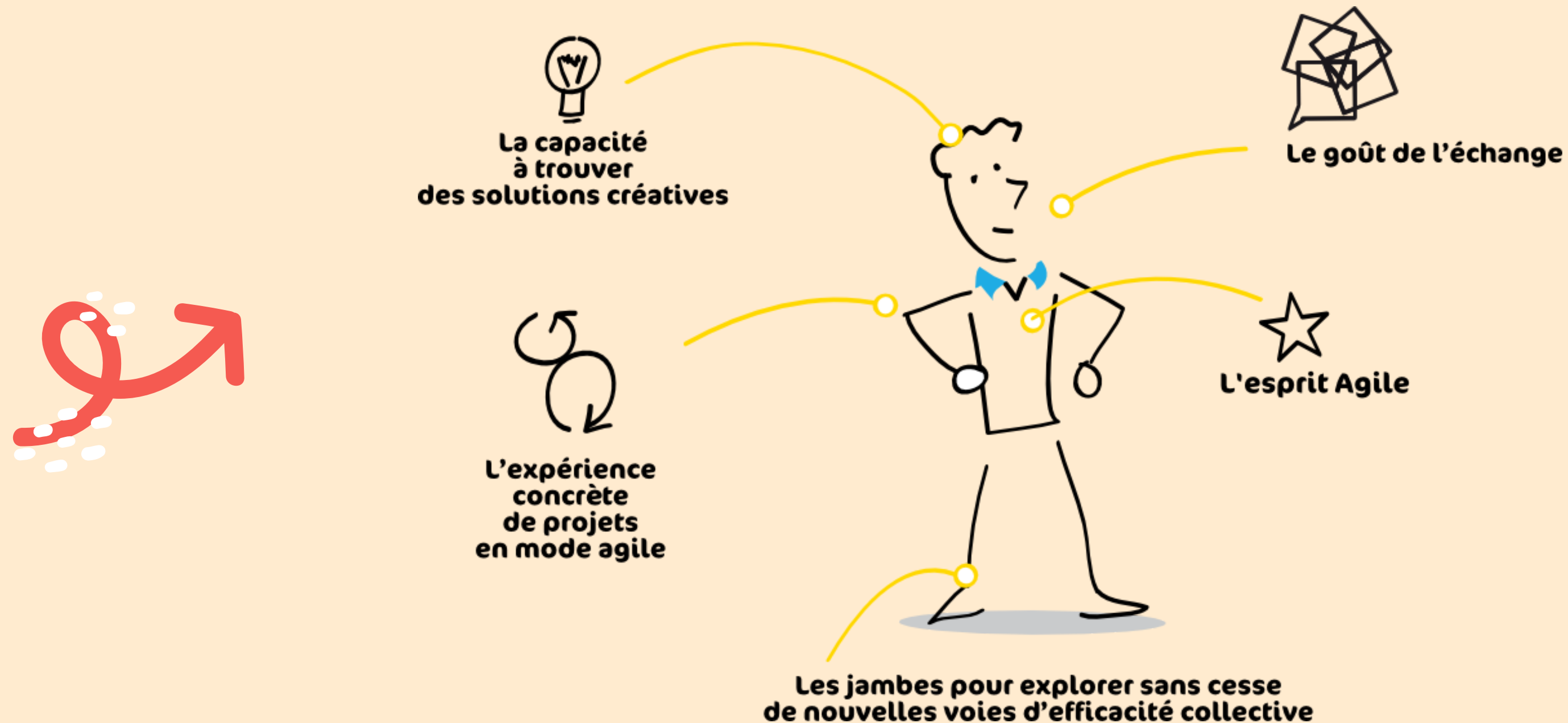


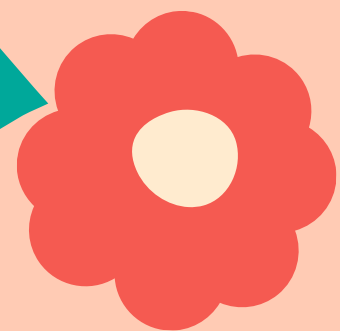
En dehors de l'équipe : les parties prenantes

- ▷ Représentants des Utilisateurs ,
- ▷ Sponsor,
- ▷ Product Managers,
- ▷ Marketing,
- ▷ Maitrise d'ouvrage,
- ▷ Direction, ...



En dehors de l'équipe : les parties prenantes





SCRUM :



Définir le besoin

Techniques pour définir le
besoin

Définir le besoin : statistiques projets

- En général, le taux d'échec des projets est élevé. De plus, plus la complexité et la taille d'un projet sont importantes, plus le risque d'échec est fort. Ce qui semble logique !
- Les gros projets ont deux fois plus de risque d'être en retard, en dépassement de budget, ou de ne pas inclure des fonctionnalités essentielles comparés aux petits projets.
- Un gros projet a 10 fois plus de probabilité d'échouer complètement, ce qui signifie qu'il sera annulé ou ne sera pas utilisé parce que son utilité est dépassée avant même sa mise en place.
- 39% de tous les projets aboutissent (livrés dans les délais, en respectant le budget, et avec toutes les caractéristiques et fonctionnalités requises),
- 43% sont livrés mais rencontrent des problèmes (en retard, dépassement de budget, et/ou avec des caractéristiques et des fonctionnalités manquantes),
- 18% échouent (soit annulés avant d'être terminés, ou livrés mais jamais utilisés).
- Seul 64% des projets atteignent leurs objectifs.

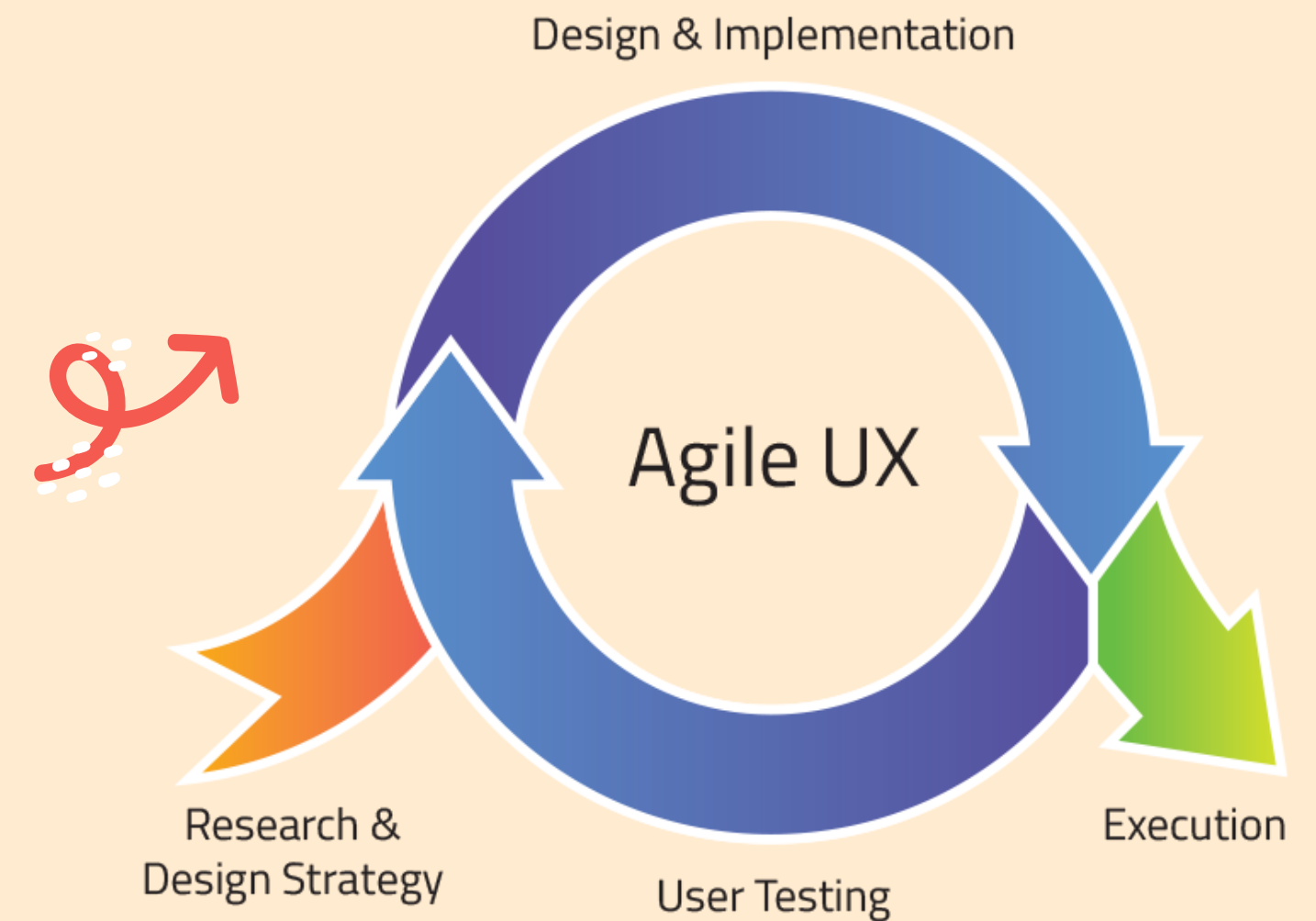
Définir le besoin : statistiques projets

- Le top 10 des causes les plus fréquentes d'échec de projet :
- Changement de priorités au sein de l'organisation,
- Exigences inadéquates,
- Changement dans les objectifs du projet,
- Risques ou opportunités non définis,
- Mauvaise communication,
- Objectifs de projet vagues,
- Mauvaise estimation des coûts,
- Mauvaise estimation du temps par tâche,
- Dépendance aux ressources,
- Mauvais management du changement.



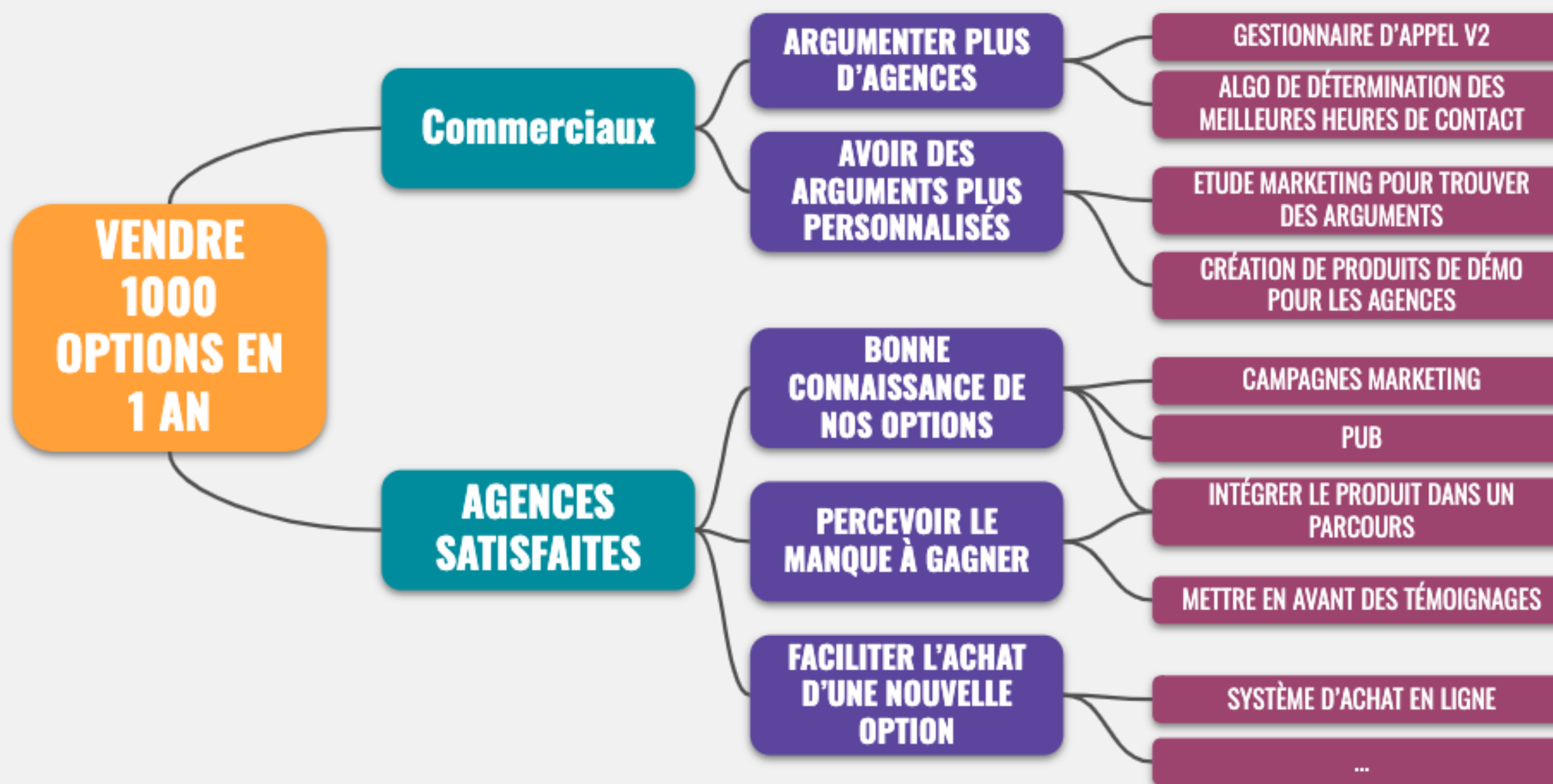
Définir le besoin : délivrer de la valeur !

- Scrum, et les méthodes agiles, maximisent la valeur en cherchant à créer des outils rapidement opérationnels.
- ▷ Estimation de la valeur : ROI, satisfaction, monnaie fictive, etc...
- Le Product Owner définit une Vision du produit, avec la participation de toute l'équipe,
- Puis il décline en User Stories organisées selon leur priorité et niveau de détail adéquat.



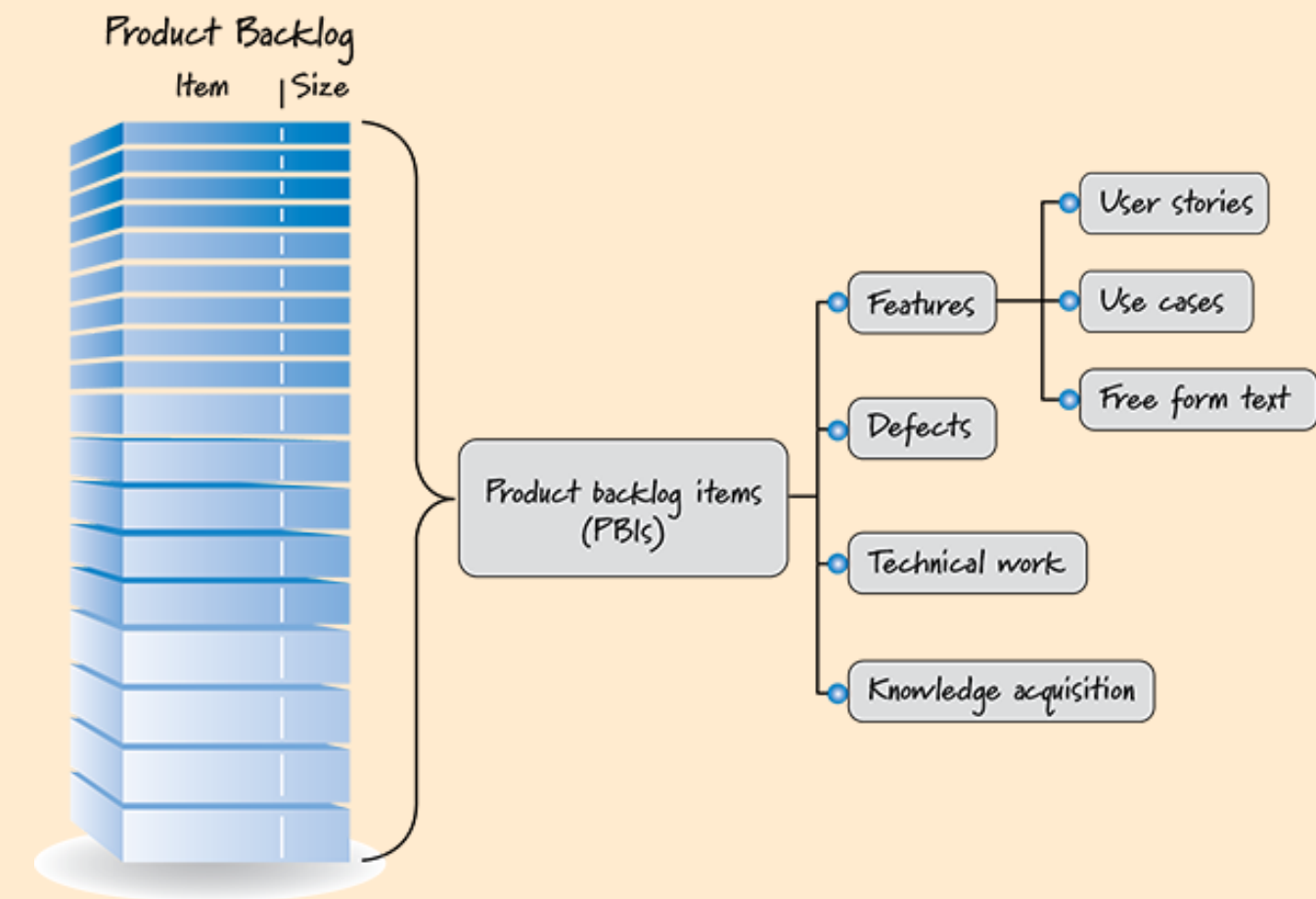
Définir le besoin : statistiques projets

OBJECTIF ACTEURS IMPACTS INITIATIVES



Le Product Backlog

- C'est une liste de fonctionnalités prioritises.
- ▷ L'ordre est stricte : un seul élément par ligne.
- Il est établi et géré par le Product Owner.
- Il met en évidence la valeur apportée, et doit être visible de tous, à tout instant !



Utilisation du Product Backlog

- Il sert de point d'entrée aux activités de planification :
 - ❖ Release planning,
 - ❖ Sprint planning.
- Il doit être à jour avant chaque activité de planification.
- Les fonctions sont triées par ordre de priorité, pour que l'équipe sache quelles sont les fonctions à intégrer dans le futur sprint.



Les User Stories

■ C'est l'outil préférentiel pour l'expression de besoin en mode Agile. Le concept provient de l'eXtreme Programming.

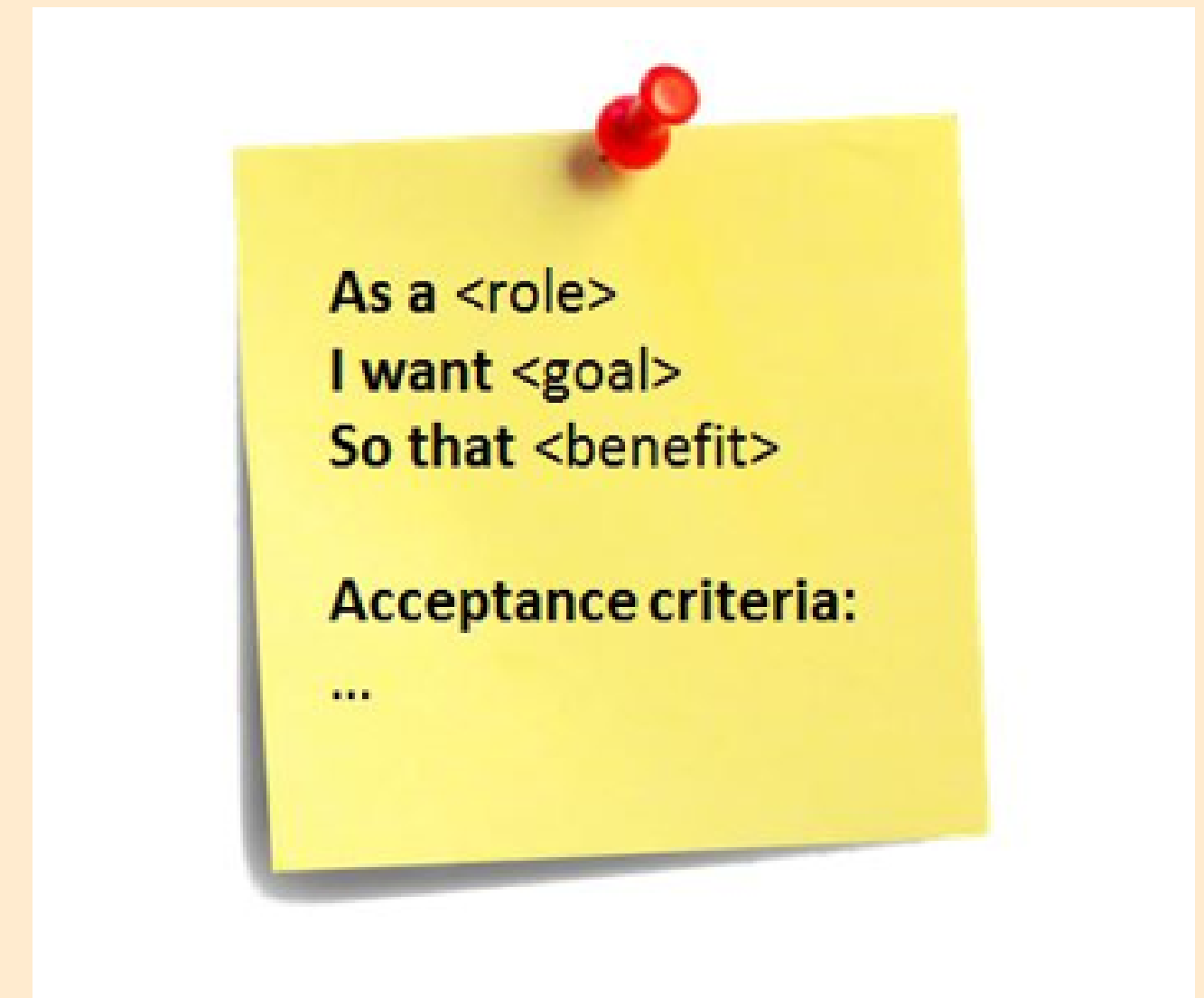
■ C'est un prétexte à la conversation.

❖ Un Backlog de User Stories s'apparente à un programme d'ateliers de discussion.



■ Souvent, pour hiérarchiser les User Stories, le PO associe une grandeur.

❖ L'importance, la valeur d'affaire, le degré de satisfaction, etc...



Les User Stories

- Utilise un formalisme simple,
- Utilise les concepts métiers sous-jacents,
- ▷ Elle est compréhensible de tous les acteurs et donne une vue métier des fonctionnalités, et n'est pas descriptive techniquement !
- Ne s'attarde pas aux détails.
- Elle se formalise ainsi :



**En tant que <rôle>,
Je veux <besoin>
Afin de <bénéfice>**

Les User Stories

■ Ron Jeffries propose les 3C pour créer les User Stories

☐ Card

■ L'histoire est écrite sur une carte de taille assez réduite

■ Ces fiches peuvent être annotées (estimation, etc.)

☐ Conversation

■ Les détails de l'histoire seront exprimés lors de conversation avec le Product Owner

☐ Confirmation

■ Des exemples (tests d'acceptation) sont consignés avec l'histoire pour valider qu'elle a été réalisée correctement



Les User Stories

- Indépendant
- Négotiable
- Valuable
- Estimable
- Small
- Testable



Exemples de User Stories

En tant qu'utilisateur, je veux pouvoir réserver une chambre d'hôtel afin de prévoir un hébergement

En tant que voyageur, je veux voir les photos des hôtels, afin de mieux affiner mon choix

En tant qu'utilisateur, je veux pouvoir annuler une réservation afin de rectifier mon erreur de réservation

En tant que voyageur régulier, je veux pouvoir re-réserver un ancien voyage, afin de gagner du temps sur mes voyages réguliers



Granularité des User Stories

■ Découpe des User Stories :

En tant qu'utilisateur, je veux pouvoir annuler une réservation afin de rectifier mon erreur de réservation



En tant qu'utilisateur premium, je veux pouvoir annuler gratuitement une réservation jusqu'à la dernière minute.

En tant qu'utilisateur non premium, je veux pouvoir annuler gratuitement jusqu'à 24h en avance.

En tant que visiteur du site, je veux un e-mail de confirmation lors d'une annulation, afin d'être sûr d'avoir une confirmation.

Critères d'acceptation

■ Découpe des User Stories :

En tant qu'utilisateur, je veux pouvoir annuler une réservation afin de rectifier mon erreur de réservation



- ☐ Vérifier qu'un membre premium peut annuler le même jour sans frais
- ☐ Vérifier qu'un membre non premium aura 10% de frais pour une annulation le même jour
- ☐ Vérifier qu'un email de confirmation est envoyé
- ☐ Vérifier que l'hôtel est notifié de l'annulation

MERCI

