

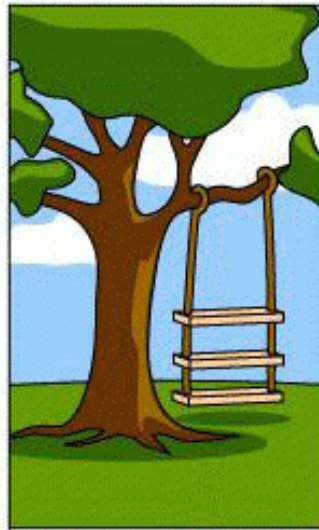
A QUOI SERT LA COLLABORATION ?

Dans son sens commun, la collaboration est un processus par lequel deux ou plusieurs personnes ou organisations s'associent pour effectuer un travail



ATELIER : Comment feriez-vous pour
collaborer pour mener à bien un projet ?





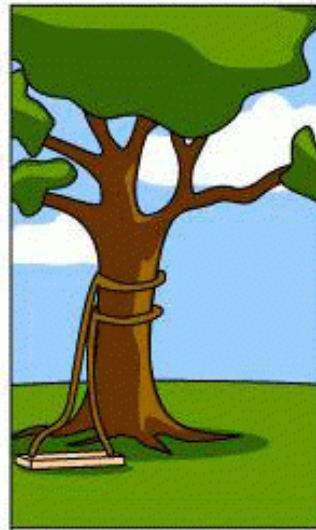
Comment le client l'a souhaité



Comment le chef de projet l'a compris



Comment l'analyste l'a schématisé



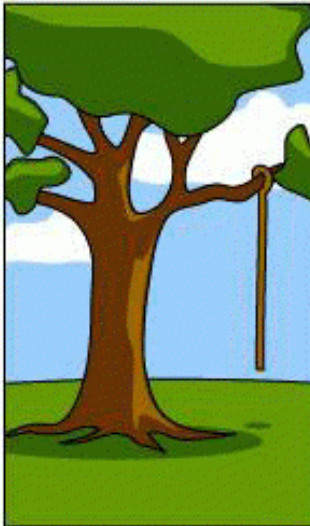
Comment le programmeur l'a écrit



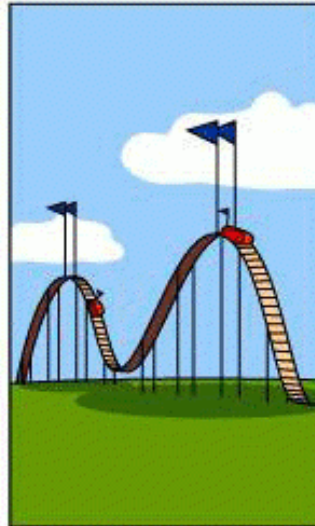
Comment le Business Consultant l'a décrit



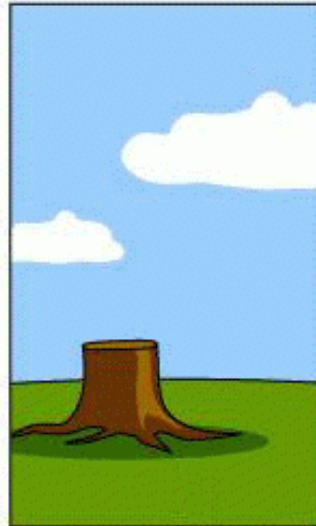
Comment le projet a été documenté



Ce qui a été installé chez le client



Comment le client a été facturé



Comment le support technique est effectué



Ce dont le client avait réellement besoin

Communiquer sur l'organisation du projet



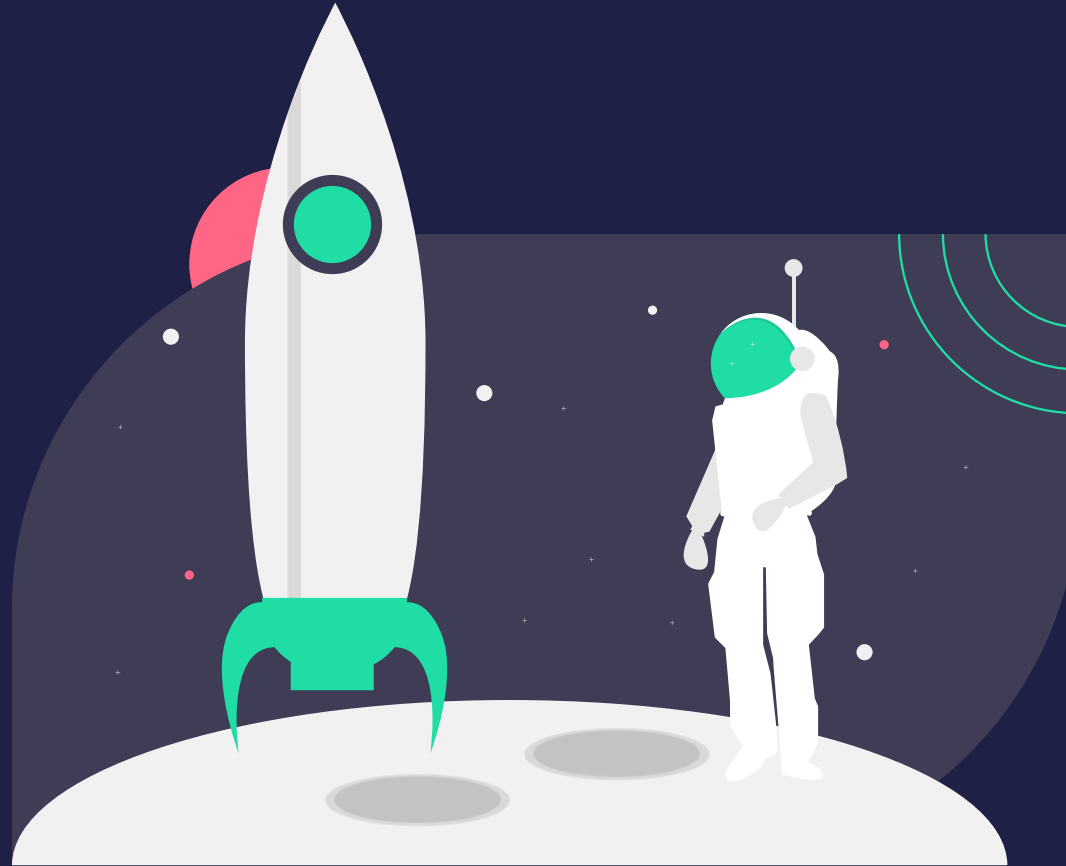
ATELIER : quelles sont les concepts clés
pour une bonne collaboration dans une
conduite de changement ?

Nous retenons les concepts principaux dans la collaboration d'un projet informatique :

- Organisation des équipes
- Planification et contrôle Analyse et conception
- Evaluation des risques
- Tests
- Gestion des incidents
- Bilan, amélioration et mutualisation

Mettre en place un processus de développement

1. SDLC
2. Lean methodology
3. Scrum
4. KPI



SDLC

Il existe différents types de **cycles de développement** entrant dans la réalisation d'un logiciel. Ces cycles prennent en compte toutes les étapes de la conception d'un logiciel.

Ce cycle de développement est aussi utilisé dans l'industrie aéronautique et spatiale pour définir des systèmes, ou des sous systèmes embarqués ou au sol qu'ils incluent ou pas de l'informatique.



Quelles sont les différents SDLC que vous connaissez ?

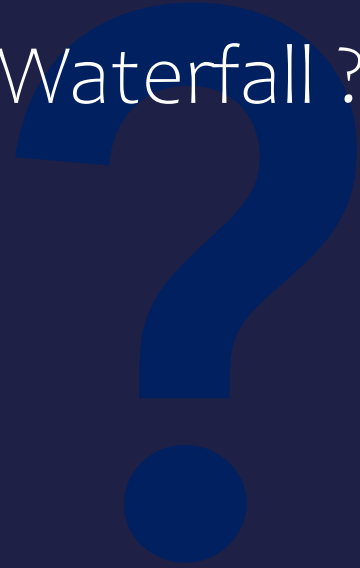


Des exemples connues sont :

- Le waterfall
- Le cycle en V
- Le cycle itératif



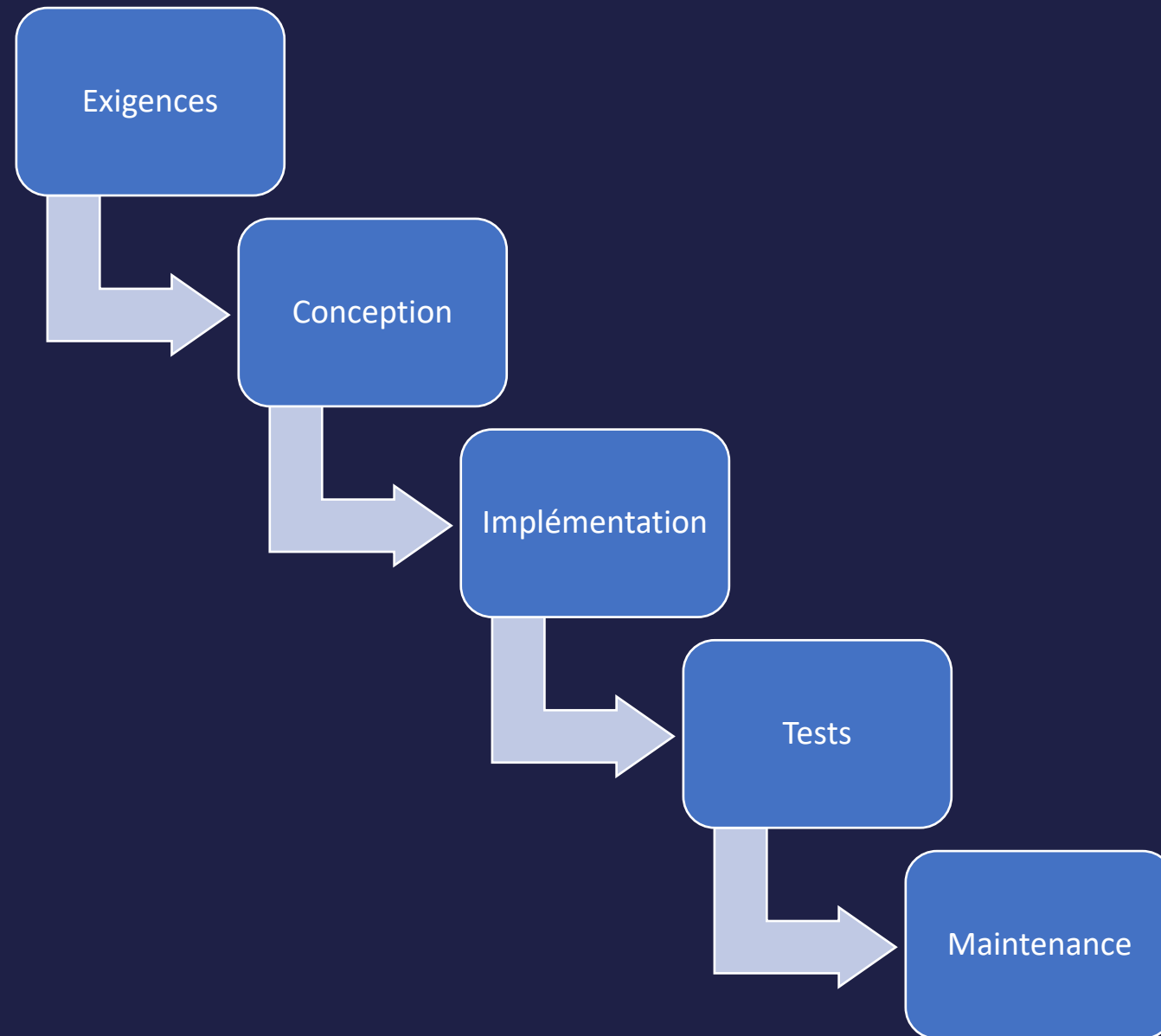
Qu'est-ce que le Waterfall ?



LE WATERFALL

Les phases traditionnelles de développement sont effectuées simplement les unes après les autres, avec un retour sur les précédentes, voire au tout début du cycle :

1. produire des livrables définis au préalable
2. terminer à une date précise
3. de ne se terminer que lorsque les livrables sont jugés satisfaisants (par des tests et une recette)





Qu'est-ce qui cloche ?





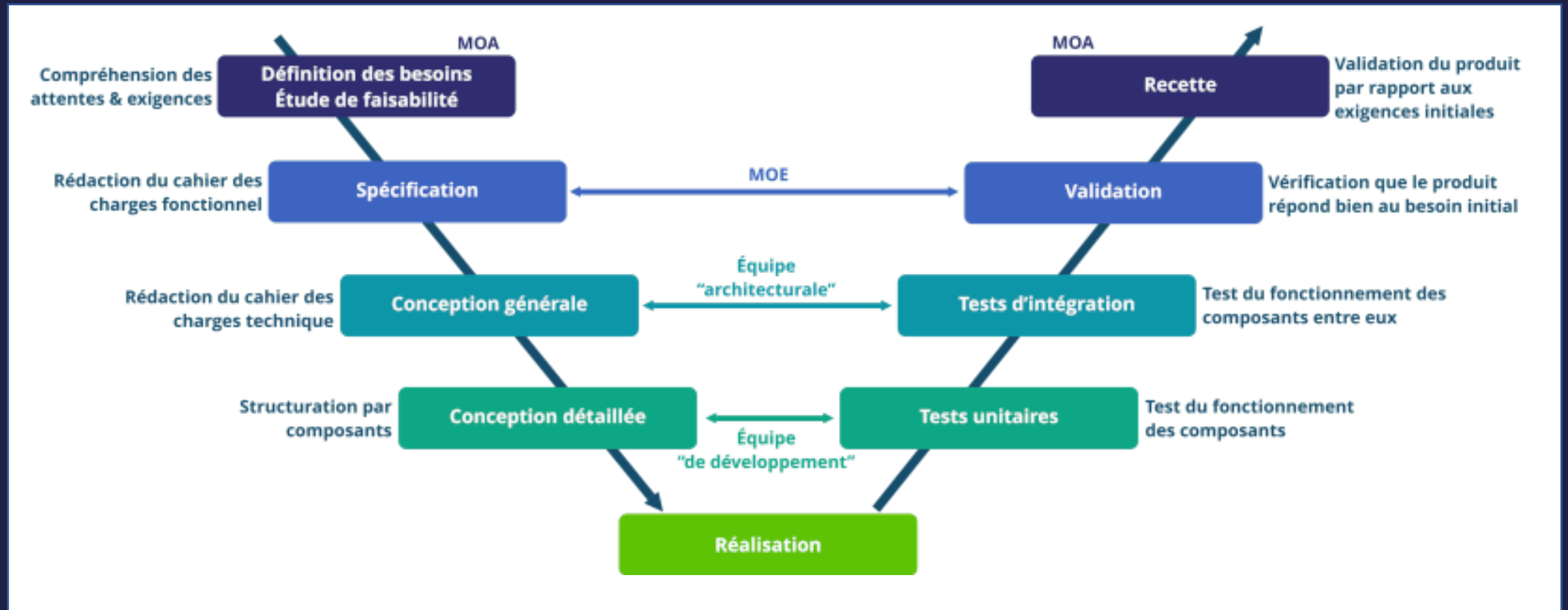
Qu'est-ce que le cycle en V ? Est-ce toujours utilisé ?



LE CYCLE EN V

Le modèle du cycle en V a été imaginé pour pallier le problème de réactivité du modèle en cascade. Ce modèle est une amélioration du modèle Waterfall qui permet en cas d'anomalie, de limiter un retour aux étapes précédentes.

- Créé suite au manque de réactivité du modèle en cascade. Il part du principe que les procédures de vérification de la conformité doivent être élaboré dès les phases de conception
- Le client doit connaître son besoin dans le détail
- Standard depuis les années 1980



Les phases de la **partie montante** doivent renvoyer **de l'information sur les phases en vis-à-vis** lorsque des défauts sont détectés afin d'améliorer le logiciel.

Il existe autant de plan de tests que de phases de qualification du produit :

- Au dossier de SPECIFICATION correspond le plan de tests de VALIDATION
- Au dossier de CONCEPTION GENERALE correspond le plan de tests d'INTEGRATION
- Au dossier de CONCEPTION DETAILLEE correspond le plan de tests UNITAIRES

Avantages	Inconvénient
<p>Meilleur temps de réaction</p> <p>Anticipation des étapes</p> <p>En cas de problème, permet de limiter le retour aux étapes précédentes</p>	<p>Conception fortement liée à la réalisation</p> <p>Moins adapté au développement logiciel</p> <p>Risque d'effet tunnel</p>



Qu'est-ce qui cloche ?



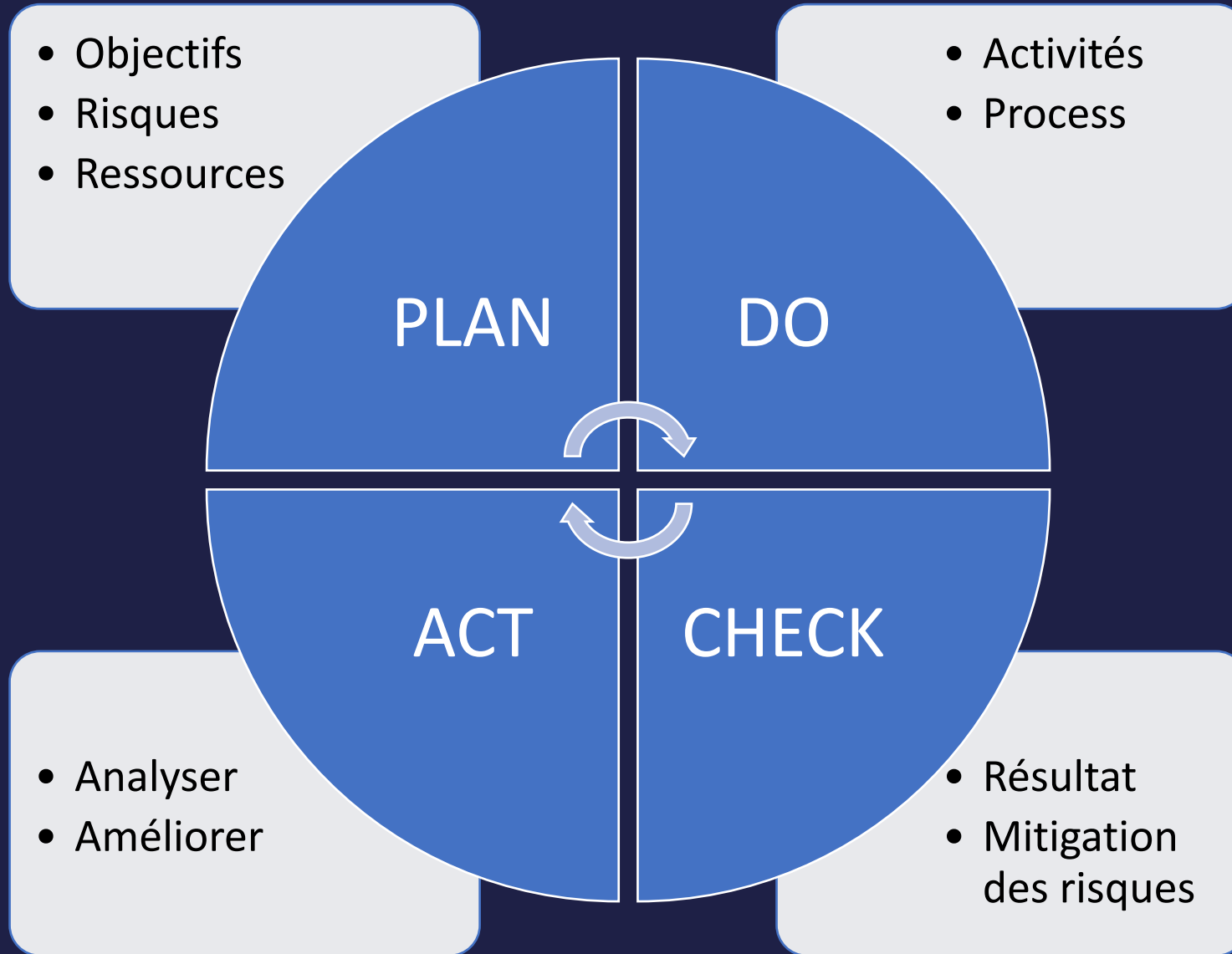


ATELIER : qu'est-ce qui définit
l'amélioration continue ?



LE CYCLE ITERATIF

On **sépare les activités des artefacts**, un artefact étant le produit issu d'une activité. Ainsi, on applique un cycle de type **roue de Deming** sur la production.





Pourquoi s'est mieux ?



Avantages	Inconvénient
<p>Cahier des charges respecté au pied de la lettre</p> <p>Validité des besoins</p> <p>Le changement est accueilli</p>	<p>Calendrier et budget demandent une expertise (moins prévisible)</p>

Le **cycle PDCA** sert à transformer une idée en action et l'action en connaissance.

1. **Plan** : préparer, planifier (ce que l'on va réaliser)
2. **Do** : développer, réaliser, mettre en œuvre (le plus souvent, on commence par une phase de test)
3. **Check** : contrôler, vérifier
4. **Act** (ou Adjust): agir, ajuster, réagir (si on a testé à l'étape do, on déploie lors de la phase act)



ATELIER : quels sont les concepts clés du
« lean » ?



LEAN METHODOLOGY

Le terme **lean** (de l'anglais lean, « maigre », « sans gras », « dégraissé ») sert à qualifier une méthode de gestion de la production qui se **concentre sur la « gestion sans gaspillage »** ou « gestion allégée » ou encore gestion « au plus juste ».

L'école de gestion lean trouve ses sources au Japon dans le système de production de **Toyota** ou SPT (en anglais Toyota Production System ou TPS).

KANBAN

Kanban est une organisation de type Juste-à-temps en fournissant l'information ponctuellement aux membres de l'équipe afin de ne pas les surcharger.

Dans cette approche, le processus complet, de l'analyse des tâches jusqu'à leur livraison au client, est consultable par tous les participants, chacun prenant ses tâches depuis une file d'attente.

Dans le cadre du développement logiciel, Kanban peut être un système visuel de gestion des processus qui indique quoi produire, quand le produire et en quelle quantité ; cette approche est directement inspirée du système de production de Toyota et des méthodes « lean ».



ATELIER : qu'avez-vous retenu de l'agilité ?



THE AGILE MANIFESTO

Notre plus haute priorité est de satisfaire le client en livrant rapidement et régulièrement des fonctionnalités à grande valeur ajoutée.

Accueillez positivement les changements de besoins, même tard dans le projet. Les processus Agiles exploitent le changement pour donner un avantage compétitif au client.

Livrez fréquemment un logiciel fonctionnel, dans des cycles de quelques semaines à quelques mois, avec une préférence pour les plus courts.

Les utilisateurs ou leurs représentants et les développeurs doivent travailler ensemble quotidiennement tout au long du projet.

Réalisez les projets avec des personnes motivées.

Fournissez-leur l'environnement et le soutien dont elles ont besoin et faites-leur confiance pour atteindre les objectifs fixés.

La méthode la plus simple et la plus efficace pour transmettre de l'information à l'équipe de développement et à l'intérieur de celle-ci est **le dialogue en face à face**.

Un **logiciel fonctionnel** est la principale mesure de progression d'un projet.

Les processus agiles encouragent un rythme de développement soutenable. Ensemble, les commanditaires, les développeurs et les utilisateurs devraient être capables de maintenir indéfiniment **un rythme constant**.

Une attention continue à l'excellence technique et à un bon design.

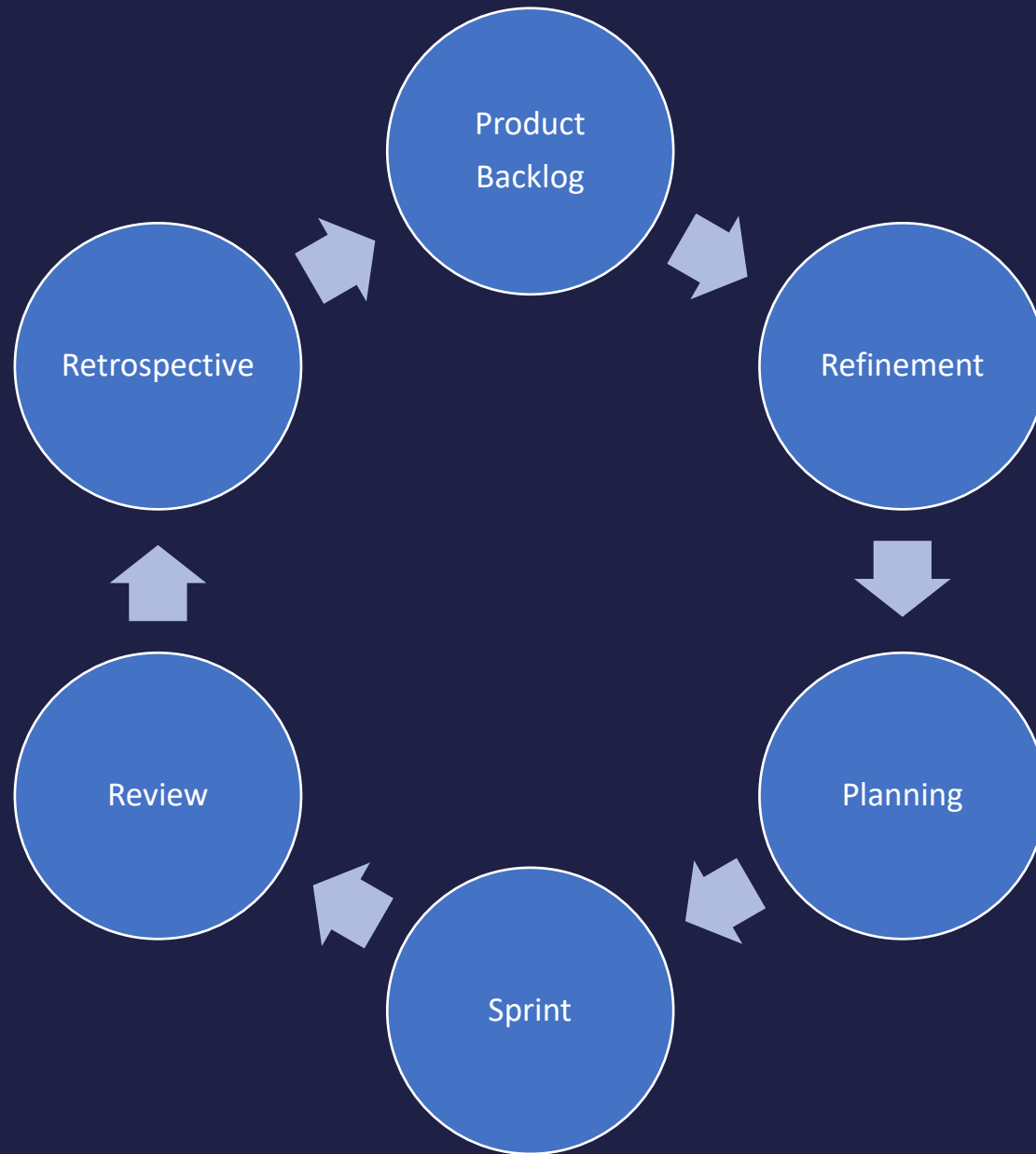
La simplicité – c'est-à-dire l'art de **minimiser la quantité de travail inutile** – est essentielle.

Les meilleures architectures, spécifications et conceptions émergent **d'équipes auto-organisées**.

À intervalles réguliers, **l'équipe réfléchit aux moyens possibles de devenir plus efficace**. Puis elle s'adapte et modifie son fonctionnement en conséquence.

SCRUM

Scrum est un framework ou cadre de développement de produits complexes. Il est défini comme un « cadre de travail holistique itératif »



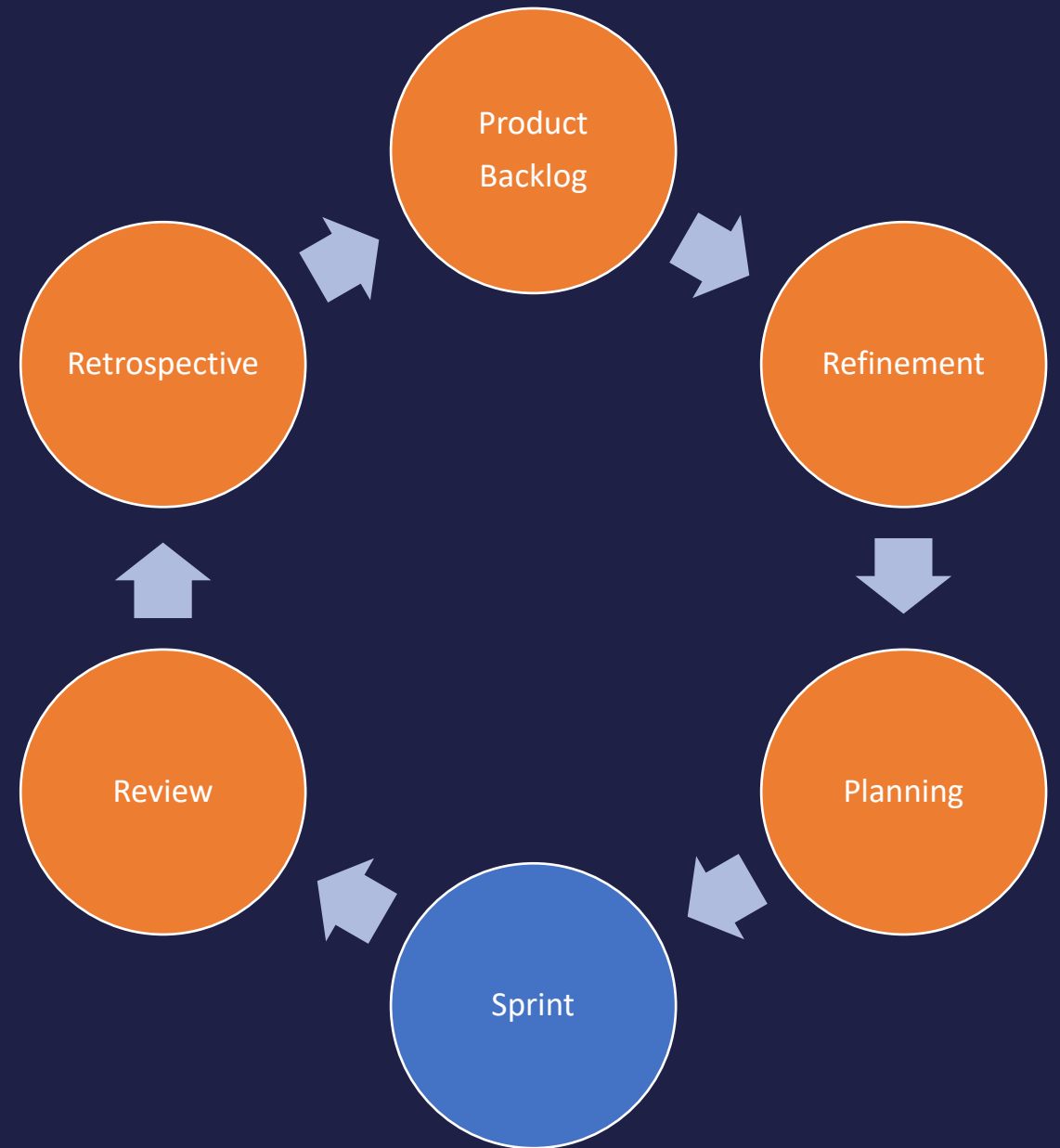
En plus de l'avancé dans la consommation (burn down) du product backlog, tous les jours un standup a lieu pour tenir ses collaborateurs à jour

Dans une équipe Scrum, les rôles associés sont les suivants :

1. Product owner
2. Scrum master
3. Developpeur(euse)s
4. Stakeholders

Le product owner est la source des exigences et de l'expression des besoins.

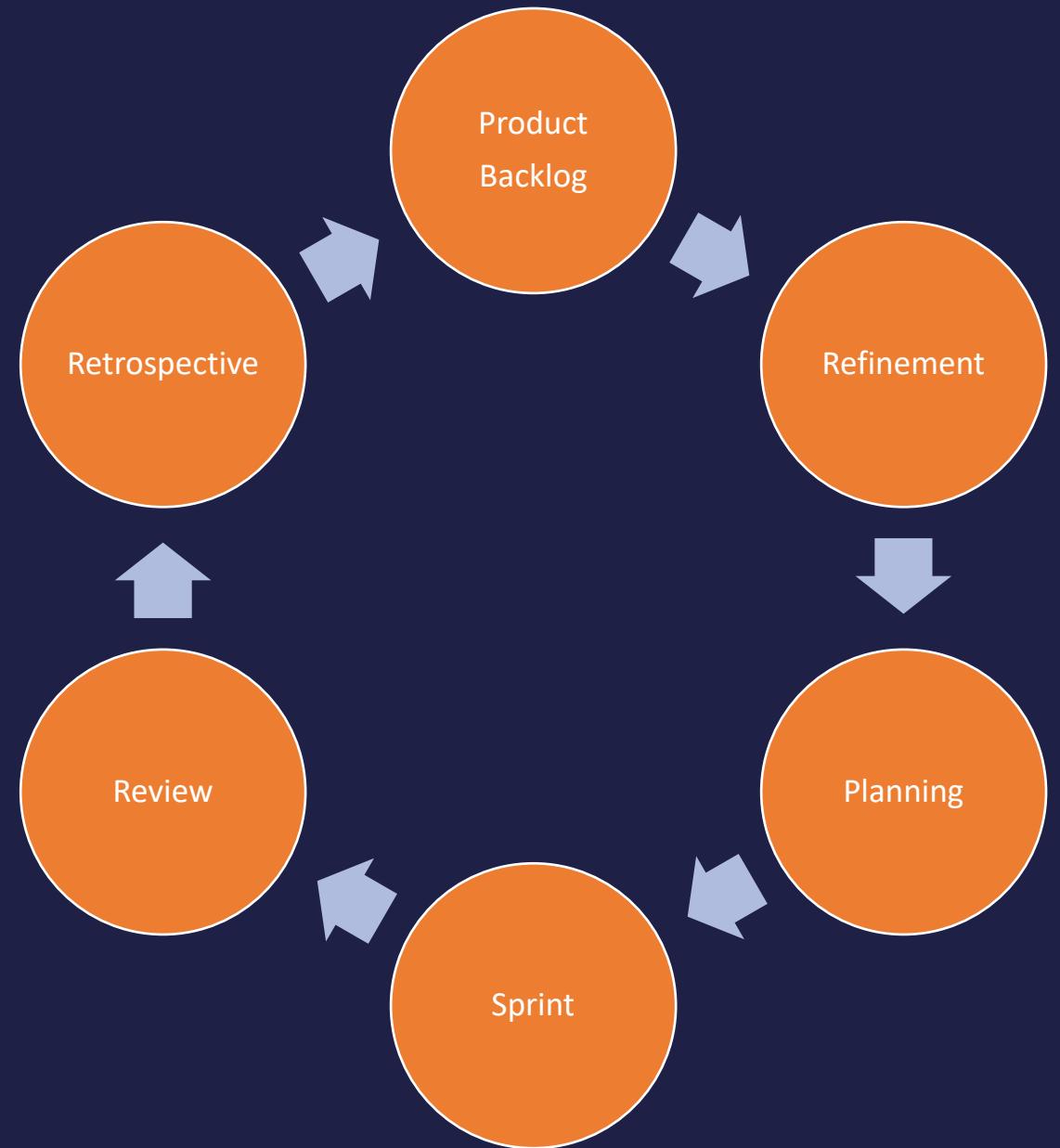
Il remplit le Product backlog avec le scrum master.



Un bon product owner est une personne qui a les idées claires et qui sait ce qu'elle veut et ce qu'elle veut en priorité

Le scrum master est un facilitateur, il s'assure que :

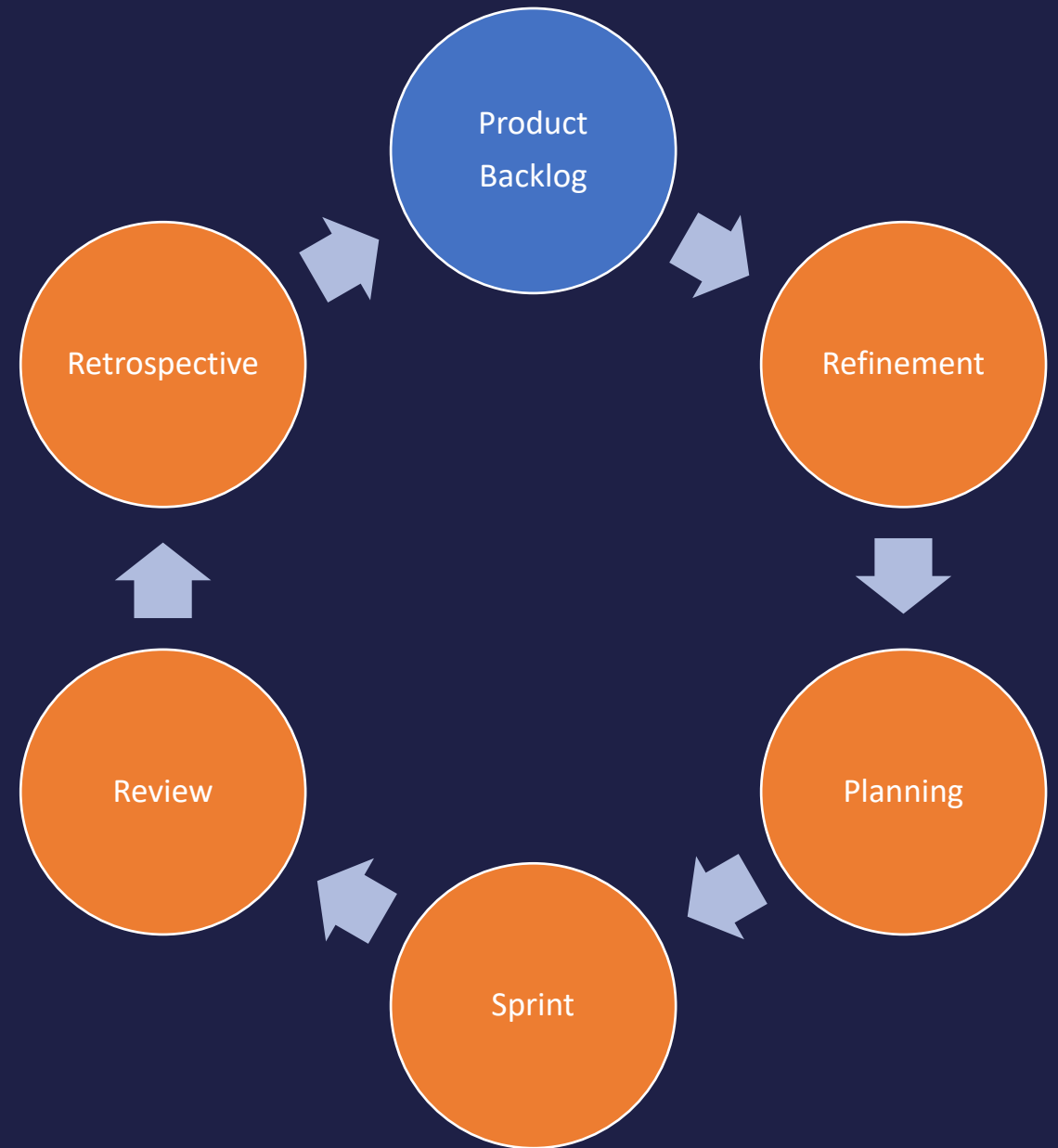
1. le framework soit bien suivi
2. la planification soit précise
3. les développeurs sont protégés des distractions extérieures



Un bon scrum master est une personne qui connaît les capacités de production des artéfacts et qui sait protéger les membres des distractions (qui prend les distractions sur lui)

Les développeur(euse)s :

1. fournissent leur expertise pour construire les incréments
2. sont auto-organisés



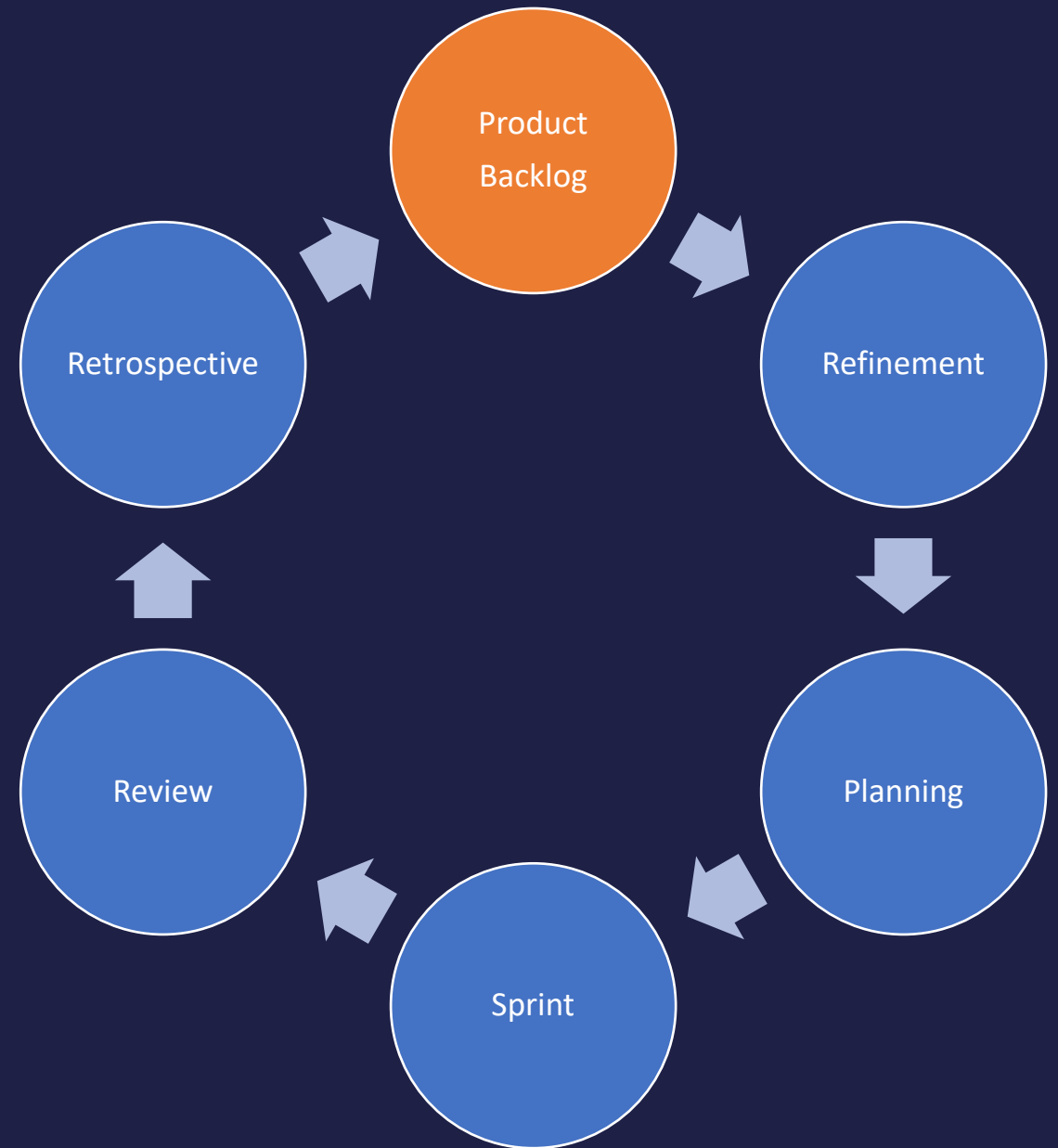
Un(e) bon(ne) développeur(euse) sait estimer le degré de complexité des features de manière cohérente et répétable

DETAILS DU CYCLE

En tant que Product Owner, définissez vos personnas, les plus-values à capturer pour chaque persona et comment il ou elle y parvient

En tant que Scrum Master, assurez-vous que le Product Owner garde une structure persona/valeur/comment

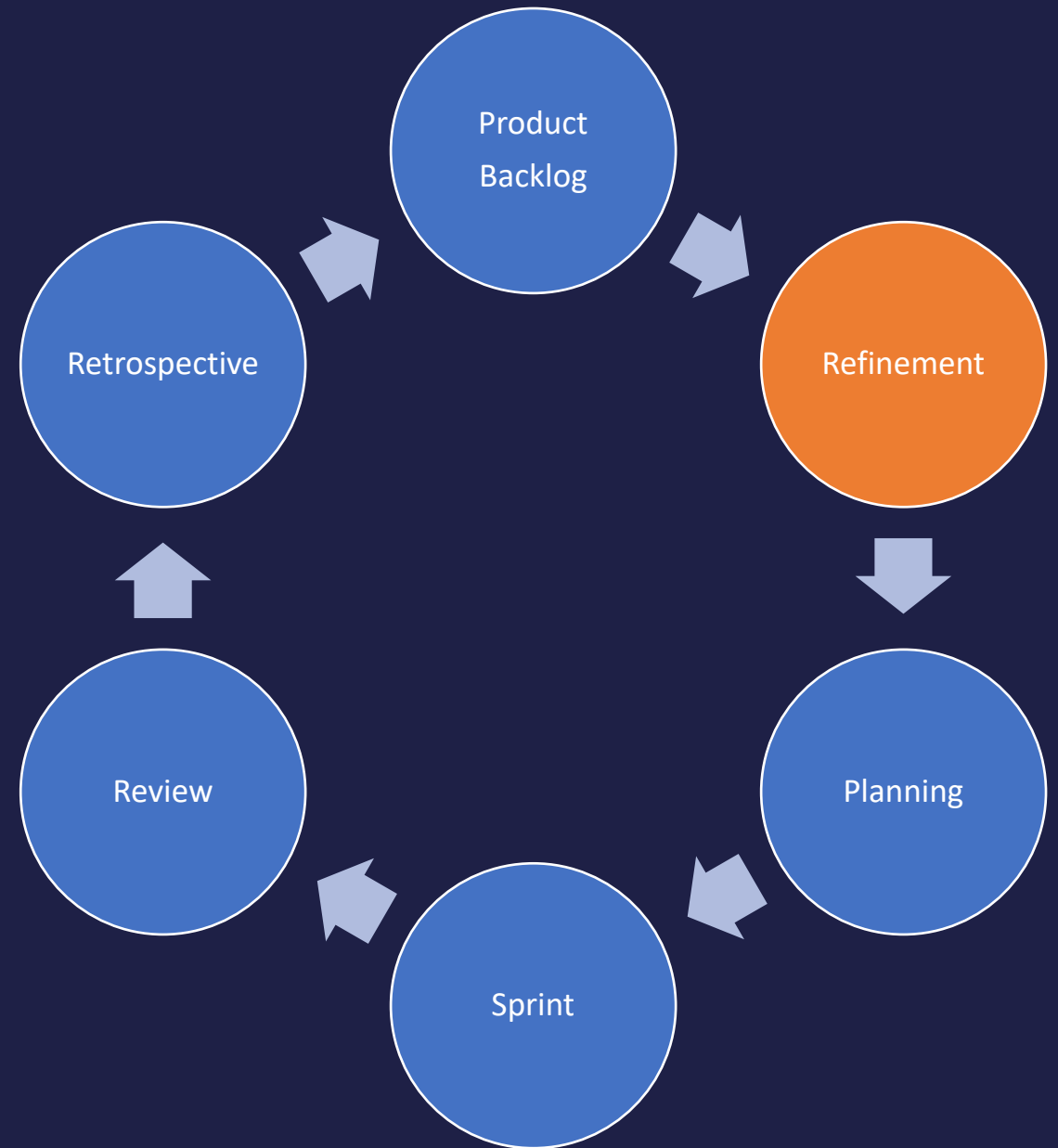
Temps : on-going



En tant que Scrum Master et développeur(euse)s :

1. Votez avec des story points (1 à 5)
2. Découpez les initiatives, les epics -> produisez des user stories

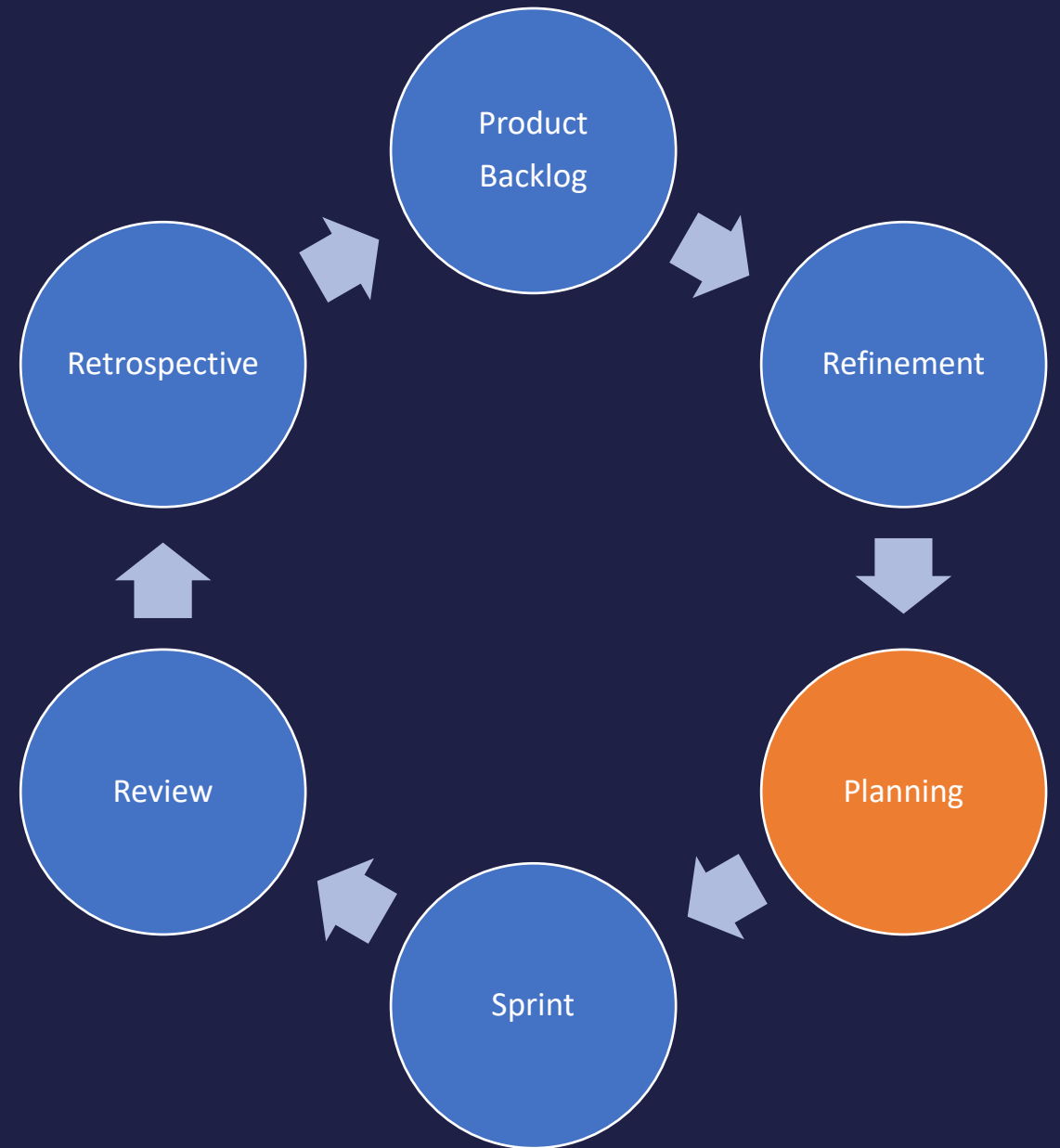
Temps : aucune contrainte



En tant que Scrum Master et développeur(euse)s :

1. Vous pouvez définir les user stories en subtasks si vous avez besoin
2. Plannifiez le sprint

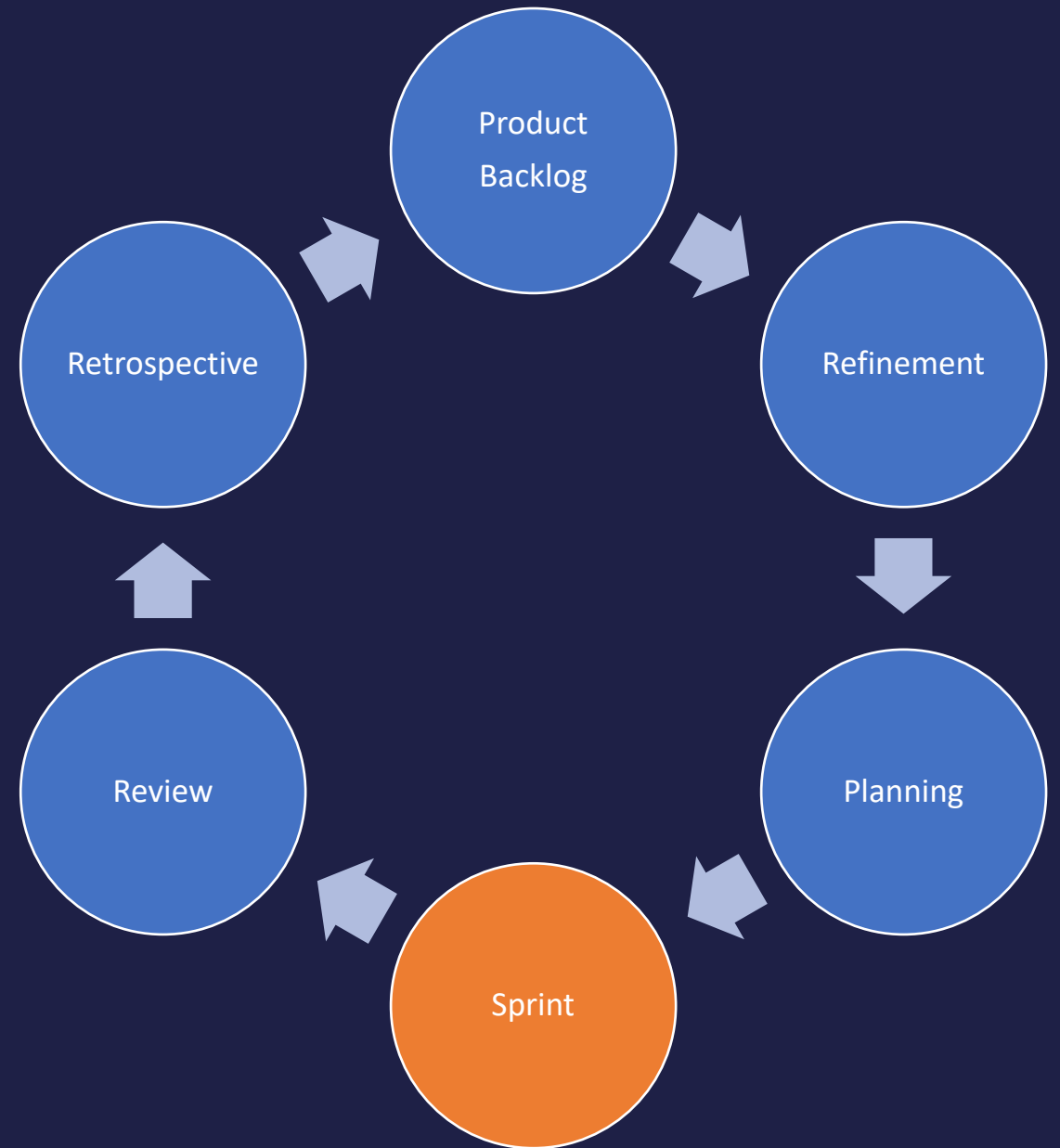
Temps : 10% du sprint



En tant que Scrum Master :
Protégez les
développeur(euse)s de toutes
distractions externes

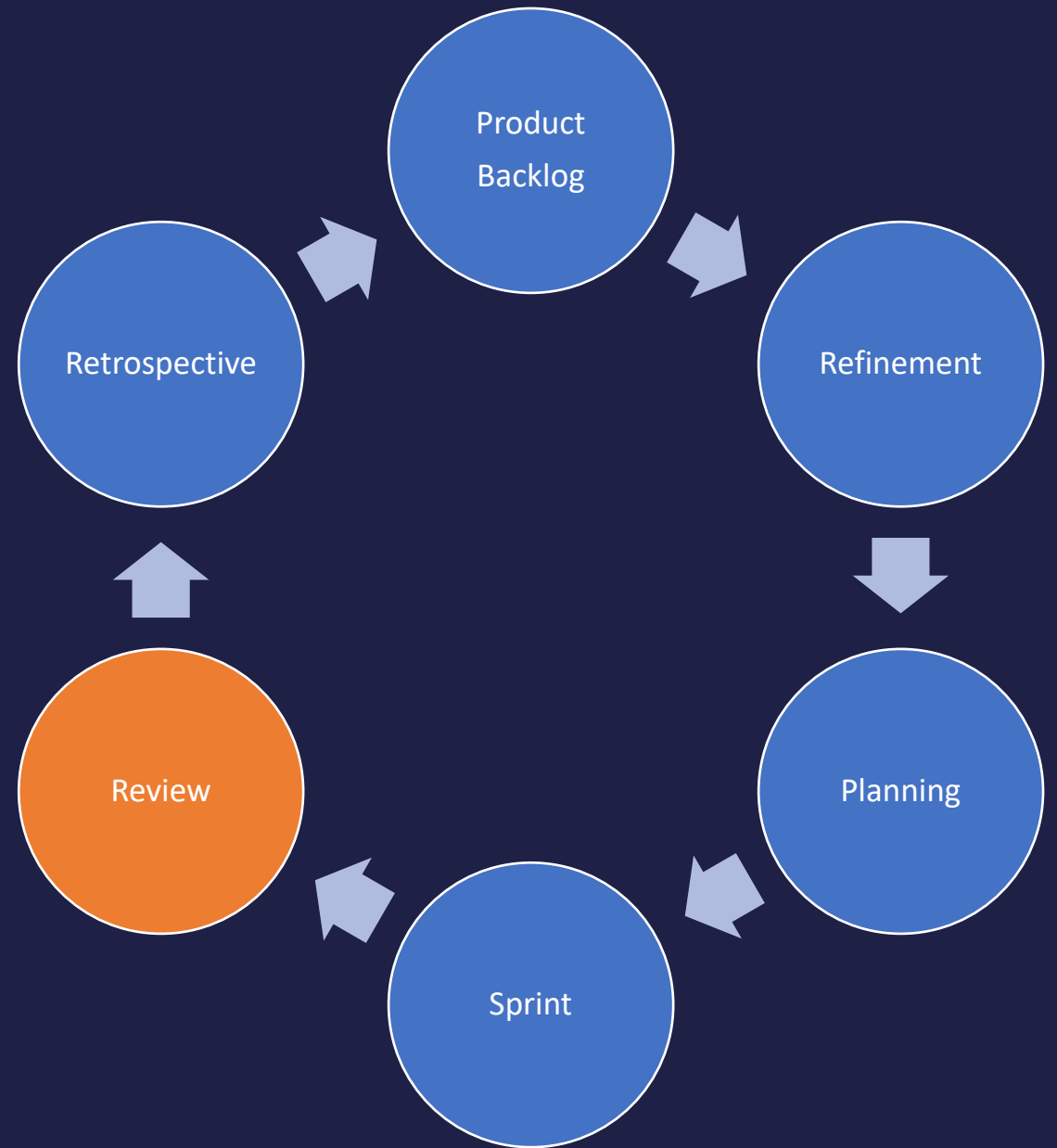
En tant que
développeur(euse)s :
Faites des standups quotidien
pour informer vos collègues

Temps : environ 1 mois



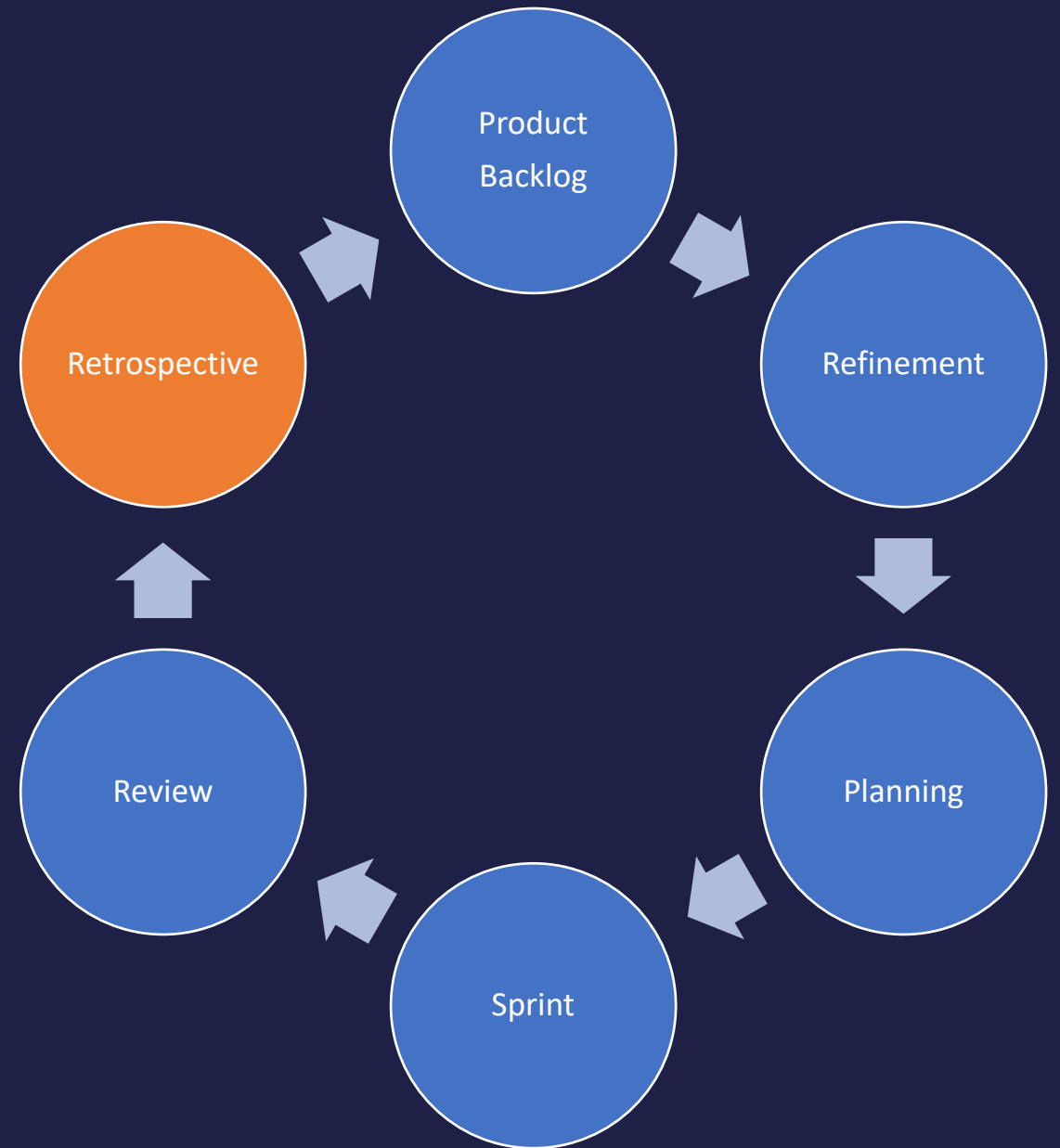
En tant que Scrum Master et développeur(euse)s :

1. Vous faites la demo de votre incrément fonctionnel
2. Vous récoltez les retours à intégrer au backlog
3. Déplacez les user stories à completed



En tant qu'équipe, vous vous améliorer :

1. Que s'est-il bien passé ?
2. Que s'est-il mal passé ?
3. Que faire différemment pour le prochain sprint ?



ATELIER : En groupe, écrivez autant de US que de participants :

Pour la création d'une application pour réserver ses plats à la cantine



ATELIER : En groupe, construisez votre équipe :

- 1) Choisir 1 PO
- 2) Choisir 1 Scrum master
- 3) Choisir 3 utilisateurs finaux
- 4) Choisir 1 graphiste
- 5) Choisir 1 responsable de la communication
- 6) Le reste seront des développeurs et des développeuses

ATELIER : En groupe, simulez des itérations Scrum :

La vélocité de chaque sprint sera fixé par votre professeur



