

# Méthodes agiles

## Méthode SCRUM

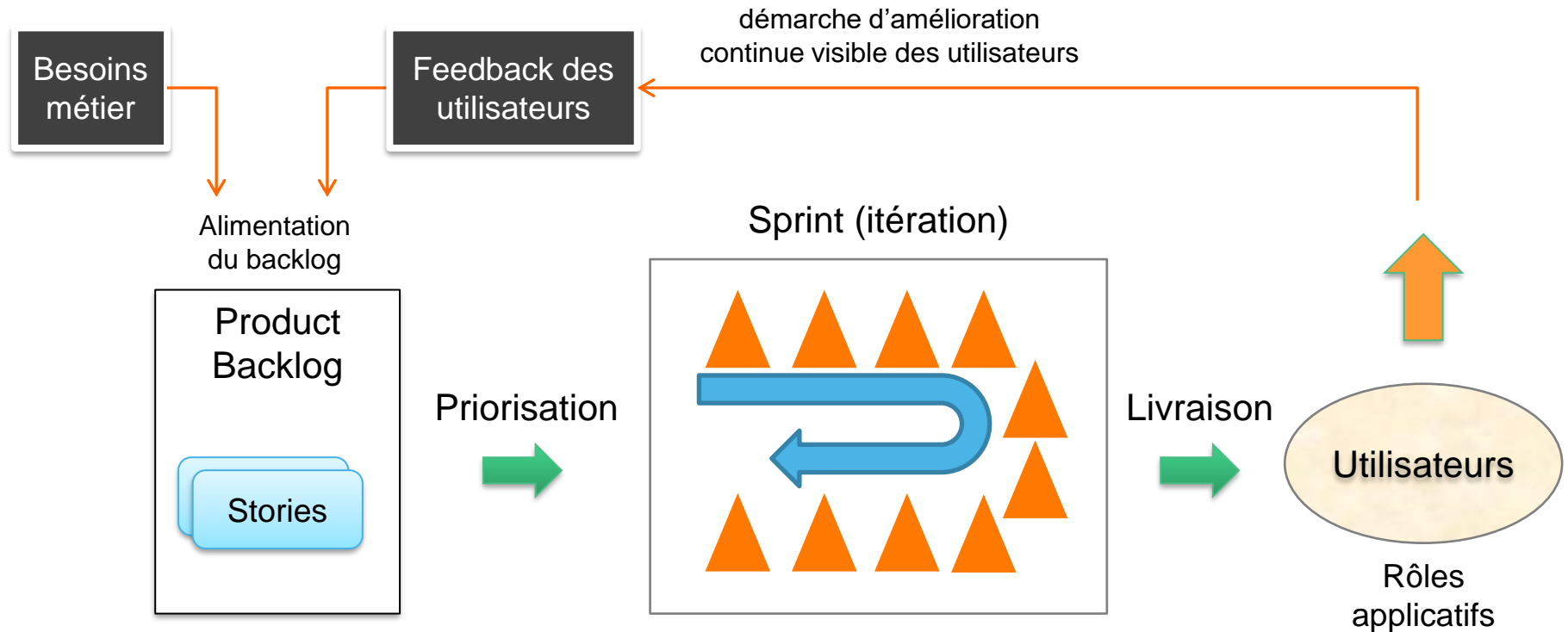


## Conduite de projet Agile

### Le Product Backlog



# Méthode SCRUM



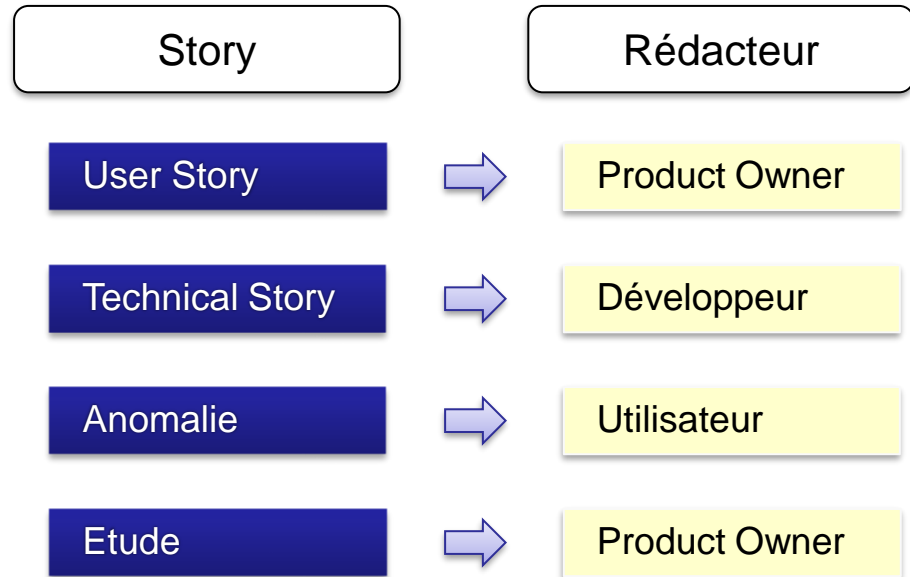
# Product Backlog

---

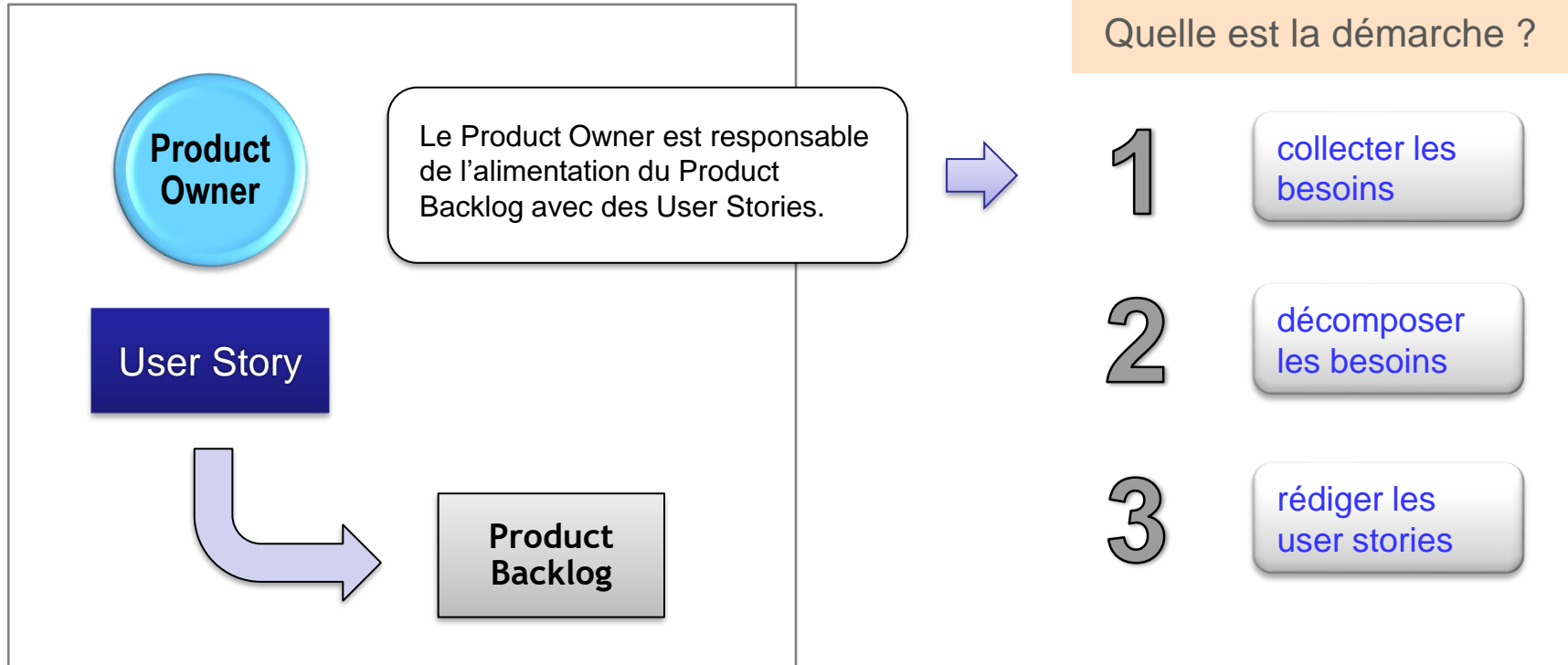
Le Product Backlog est le référentiel des travaux qui restent à faire.

Le Product Backlog accueille sous la forme de stories **tous les travaux** qui seront confiés aux développeurs.

**Tous les travaux** doivent être enregistrés dans le Product Backlog car leur existence doit être déclarée afin de permettre leur évaluation et leur priorisation avant de les engager dans une itération à venir.

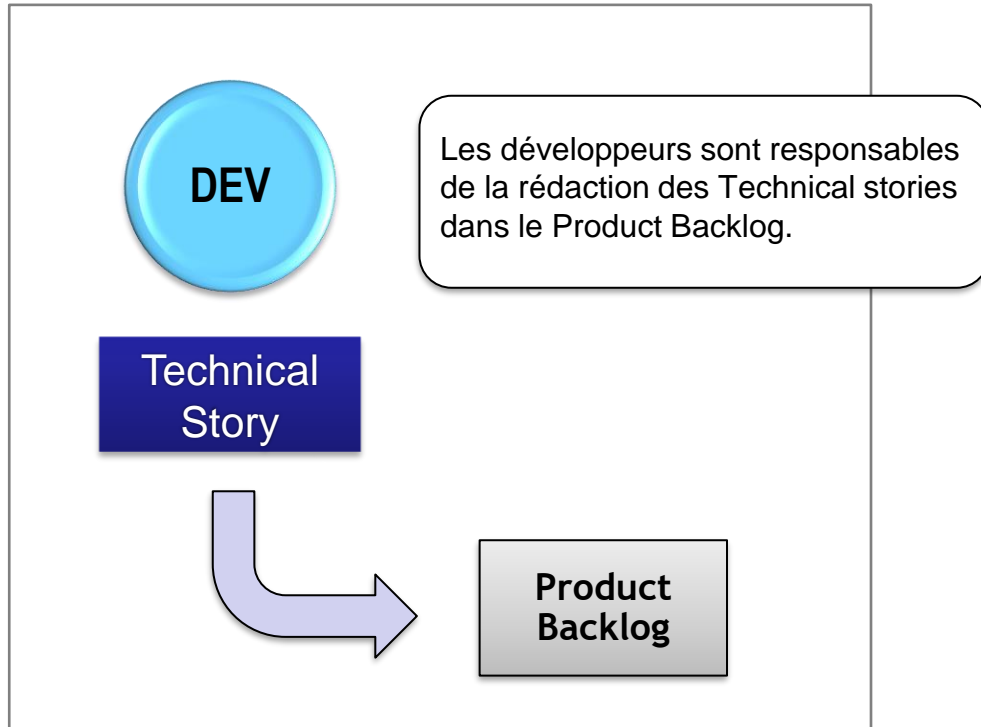


# User stories



Les User Stories du Product Backlog sont les stories les plus nombreuses.

# Technical stories



Quelle est la démarche ?

1

définir le  
besoin

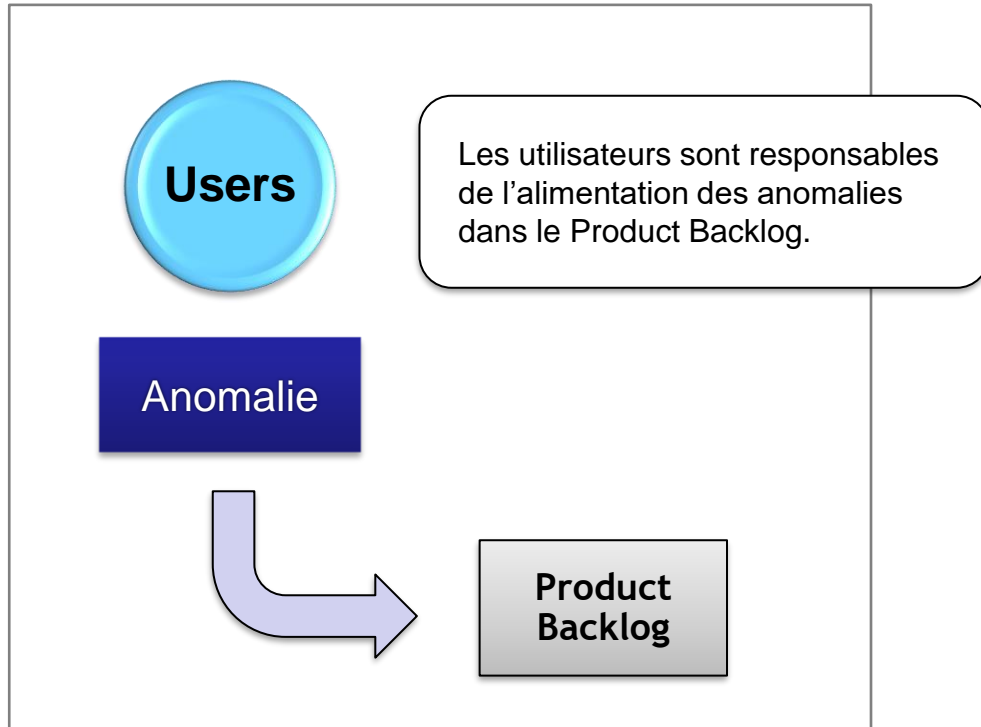
2

rédiger la  
technical story

## Exemples

- upgrade d'un logiciel,
- ajout d'une plateforme,
- accroissement d'un filer,
- etc.

# Anomalies - Defects



Quelle est la démarche ?

1

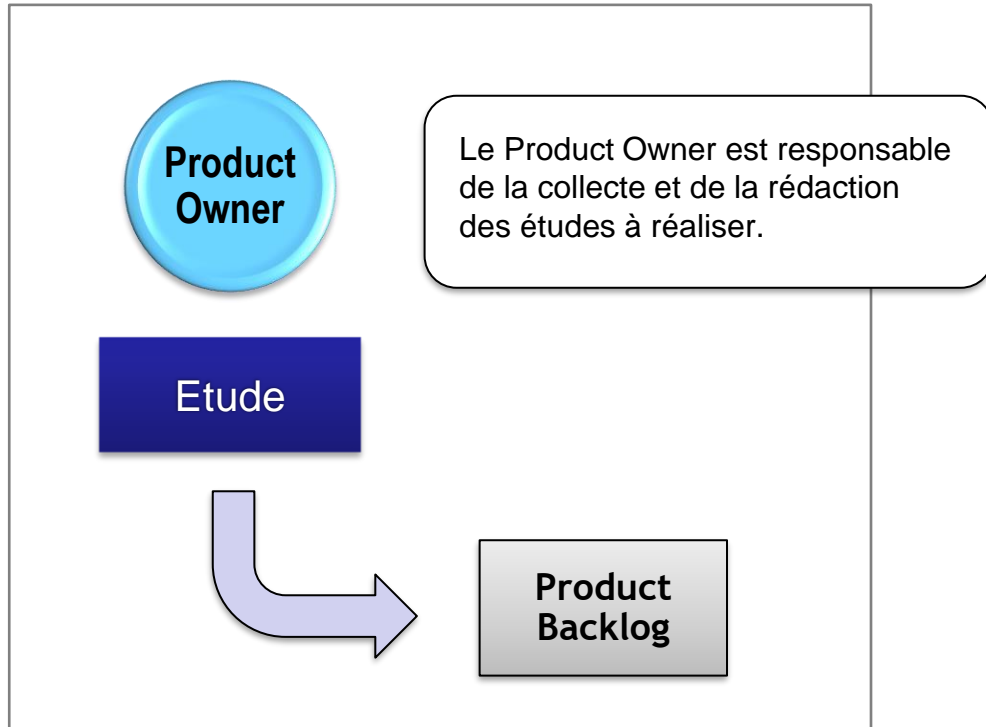
identifier le  
problème

2

rédiger  
l'anomalie

Le Product Owner ou un développeur peuvent apporter leur soutien à la rédaction de la story Anomalie.

# Etudes



Quelle est la démarche ?

1

collecter le  
besoin

2

rédiger l'étude

L'étude demandée par le Product Owner est réalisée par les développeurs. La proposition de solution permettra au PO de prendre une décision et de rédiger les user stories associées.

# Ce qu'il faut retenir

---

1

User  
stories

Les **User Stories** définissent les incréments fonctionnels du produit. Elles suivent les règles agiles (décomposition simple, mémo INVEST, story READY) et incluent les règles de gestion validées par les métiers. Les User Stories sont priorisées par le Product Owner et évaluées par les développeurs.

2

Technical  
stories

Les **Technical Stories** sont soumises par l'équipe de réalisation. Elles représentent des travaux utiles aux environnements et/ou aux plateformes du projet. Elles permettent également de réaliser l'installation des nouveaux logiciels et des outils nécessaires au projet.

3

Anomalies

Les **Anomalies** sont soumises par les utilisateurs afin de corriger les défauts observés sur l'application en production. Les anomalies sont priorisées en fonction de leur sévérité (Critique, Majeure, Mineure) et du niveau d'effort associé.



# Les études

---

1

Le démarrage d'un projet agile doit s'accompagner de l'identification et de la mise en œuvre des études visant à apporter des réponses et des propositions de solution aux problématiques du Produit.

2

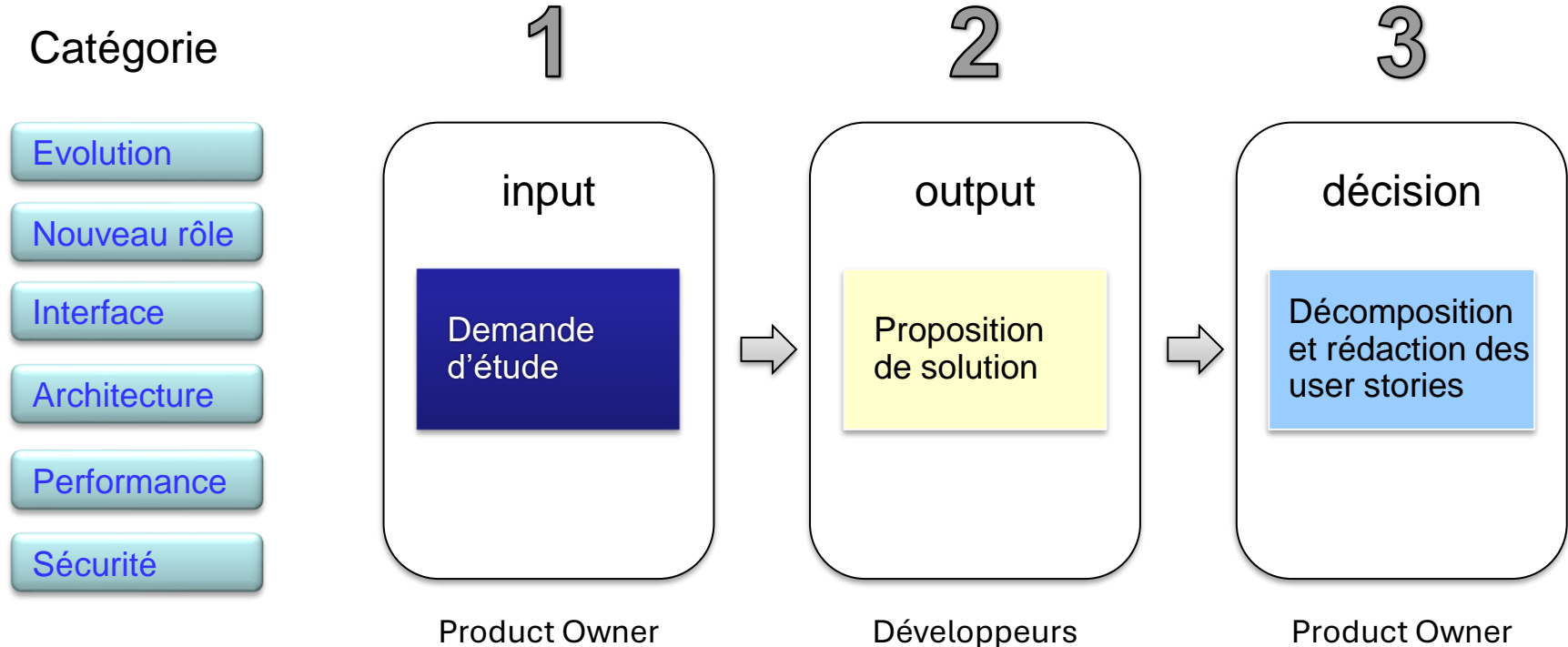
Les études sont enregistrées dans le Product Backlog par le Product Owner et représentent un niveau d'effort pour l'équipe de développeurs au même titre que les autres stories.

3

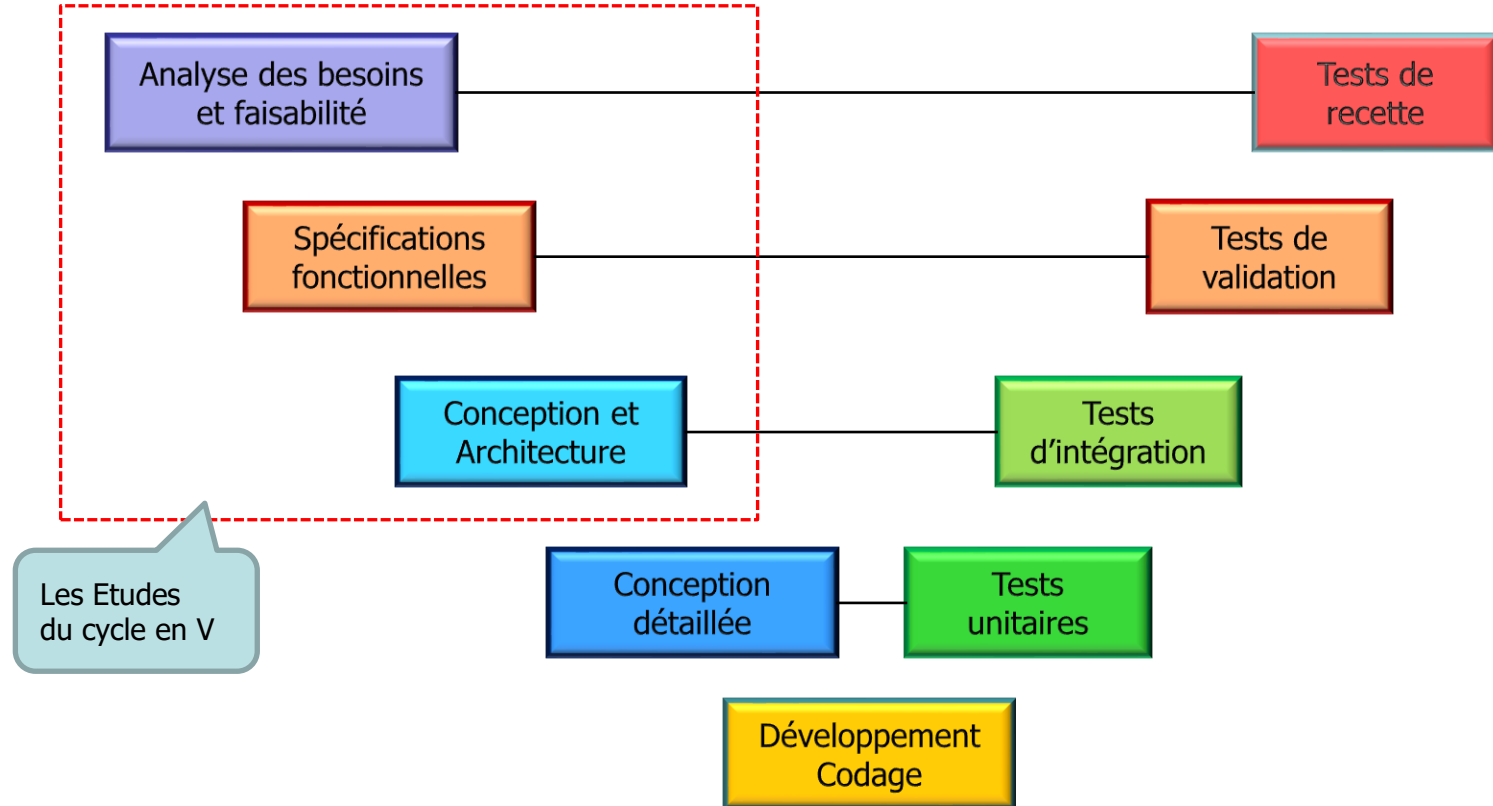
En vie de solution, l'introduction d'une nouvelle interface, d'un nouveau rôle utilisateur, ou le traitement d'un temps de réponse dégradé requièrent la rédaction d'une story de type Etude.

# Les études

## Comment ça marche ?



# Les études - Projet classique



# Les études - Projet classique

---



Les études correspondent à un investissement important en termes de budget (k€).

Les études correspondent à un investissement significatif en termes de valeur métier et de connaissances.

Les études sont le socle qui porte l'engagement de résultat du prestataire.

L'introduction de changements imprévus remet en cause les études initiales et donc l'obligation de résultat du prestataire.

Le calendrier et le budget du projet classique peuvent être significativement impactés par l'introduction de changements.

# Les études - Projet agile

---

Les études doivent être identifiées à partir de la Vision du Produit définie par le Client.

Elles peuvent être découvertes à partir des problématiques embarquées avec le Produit conformément à la Vision du Produit validée par le client.

Les études doivent prendre en compte les besoins transverses (NFR) rédigés dans les Critères du Done des User stories du Product Backlog.

Les principales catégories d'études sont les suivantes :

Evolution

Architecture

Nouveau rôle

Performance


Interface

Sécurité

NFR : Non-Functional Requirements  
Il s'agit des contraintes transverses qui  
s'applique au produit dans son ensemble.

# Les études - Projet agile

---




La vie de solution correspond à la période du projet qui intervient après la livraison de la première version du Produit.

La première version du Produit livré en production aux utilisateurs correspond au MVP (Minimum Viable Product) du Produit. Cette version inclut le noyau critique des fonctionnalités et des besoins transverses nécessaires aux utilisateurs du Produit.

En vie de solution, les retards d'identification des études à lancer sont autant de risques de ne pas accomplir la Vision du Produit dans le cadre du calendrier attendu par le client.


# Ce qu'il faut retenir

---

- 
- Le Product Backlog est créé et alimenté par le Product Owner.
  - Les stories du Product Backlog sont classées par ordre de priorité.
  - Le Product Backlog s'enrichit de nouvelles stories avec l'avancement de la découverte et de l'exploration fonctionnelle du Produit.
  - La priorisation du Product Backlog est à faire en amont de la préparation de la prochaine itération (avant le Sprint Planning).
  - Le volume des stories engagées avec la prochaine itération (Sprint Backlog) est assujetti à la vélocité de l'équipe de réalisation.

# Ce qu'il faut retenir

---

- 
- L'alimentation continue du Product Backlog pendant toute la durée du projet agile conduit le Product Owner à réaliser de nouvelles actions de priorisation à valider avec le client.
  - Ces actions de priorisation conduisent à revisiter le plan de livraison des versions et des releases afin que le nouveau calendrier soit en accord avec la trajectoire fonctionnelle validée par le client avant d'être partagée avec toute la communauté agile du projet.
  - Conclusion : la trajectoire prévisionnelle d'un projet agile n'est pas respectée. Il s'agit du fonctionnement normal d'un projet agile qui intègre par définition les changements qui modifient sa trajectoire fonctionnelle.



# Méthodes agiles

**Merci**

