

Méthode Agile

Mercredi 04 Décembre 2019



QU'EST-CE QUE L'AGILITE ?

QU'EST-CE QUE L'AGILE ?

Première communication sur des notions agiles : 1986

- ✓ Article dans le magazine Harvard Business Review
- ✓ Comment les entreprises Honda, Canon, Fuji-Xeros réalisent des résultats de classes mondiales sur des équipes à géométrie variables avec des développements produits en tâches parallèles ?

QU'EST-CE QUE L'AGILE ?

- ✓ Méthode **d'organisation et de réalisation** d'un projet (IT ou autre)
- ✓ Forte **implication** de **l'ensemble** des **intervenants** du projet
- ✓ Principe de base : le **périmètre** du produit à réaliser est **non déterminé et non figé**
- ✓ **Acceptation** par tous les acteurs que les **tâches** réalisées sont **imparfaites**
- ✓ Droit à **l'erreur**
- ✓ Prise en compte de **changements** fonctionnels **rapidement** et **livraison rapide** de ces changements aux clients
- ✓ Correction et amélioration **par ajout successif** (itération)

QU'EST-CE QUE L'AGILE ?

2001 : rédaction du **manifeste agile**

→ 4 valeurs fondamentales

→ 12 principes clés

LES 4 VALEURS FONDAMENTALES DE L'AGILE



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



Des logiciels opérationnels plus qu'une documentation exhaustive



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



La collaboration avec les clients plus que la négociation contractuelle

LES 12 PRINCIPES FONDAMENTAUX (1/2)



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 opérationnel avec des cycles de quelques semaines à quelques mois et 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 ils 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.

LES 12 PRINCIPES FONDAMENTAUX (2/2)



Un logiciel opérationnel est la principale mesure d'avancement



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 une bonne conception renforce l'Agilité.



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



À intervalles réguliers, l'équipe réfléchit aux moyens de devenir plus efficace, puis règle et modifie son comportement en conséquence.



Les meilleures architectures, spécifications et conceptions émergent d'équipes autoorganisées.



LES DIFFERENTES METHODES AGILES

- ✓ Rapid Application Development (RAD, 1991)
- ✓ Dynamic systems development method (DSDM, 1995, consortium anglais commercialisant le RAD)
- ✓ Scrum (1996)
- ✓ Extreme programming (XP, 1999)
- ✓ Adaptive software development (ASD, 2000)
- ✓ Développement basé sur les fonctionnalités (FDD pour Feature Driven Development, 2003)
- ✓ Behavior-driven development (BDD, 2003)
- ✓ Crystal clear (2004)



PROJET SIAAP : Mise en place de la méthode SCRUM



- I. LA METHODE SCRUM
- II. VOCABULAIRE
- III. LES ACTEURS
- IV. ORGANISATION PROJET
- V. LES RITUELS

I. LA METHODE SCRUM

Mercredi 04 Novembre 2019



CE N'EST PAS

Une solution à appliquer et suivre passivement

Une baguette magique à la résolution de tous les problèmes

C'EST

Modèle visant à aider à organiser le travail de réalisation d'un produit

Structure le travail d'équipes auto-organisées sur des courtes itérations de temps

Maximise : le contrôle, la flexibilité, la qualité et le retour sur investissement



Scrum : terme venant du rugby désignant une mêlée

Repose sur les valeurs suivantes :

- Honnêteté
- Ouverture aux autres
- Courage
- Respect
- Concentration
- Confiance
- Responsabilité
- Coopération

II. VOCABULAIRE (Artéfacts)

Mercredi 04 Novembre 2019

AQUEDI

GRUPE
studia

STRUCTURE (1/3)

PRODUCT BACKLOG (Carnet de produit)

Liste des fonctionnalités à développer ou des tâches à paramétrer nécessaire à la réalisation du produit

USER STORY (US)

Description d'un besoin fonctionnel du produit à réaliser.

Une US doit être :

- Estimable
- Testable
- Non dépendante d'une autre fonctionnalité
- Négociable

Elle est constituée

- un nom
- une description fonctionnelle détaillé permettant à une membre de l'équipe de développement de comprendre et réaliser facilement la fonctionnalité à développer
- un ou de plusieurs tests d'acceptance
- Une estimation de charge

Formalisation d'une US

- En tant que [ROLE UTILISATEUR] je [VERBE D'ACTION] [ACTION A REALISER] afin de [BENEFICE DE L'ACTION]

STORY POINT

Correspond à l'estimation de la carte. Elle peut-être en point de complexité, en jour, en heure, en minute, ect...

DEFINITION OF DONE (DOD)

Doit répondre à la question « Est-ce que cette User Story a été implémentée ? »

Ensemble de critère permettant de s'assurer de la qualité de l'implémentation d'une US.

Objectifs :

- *Donner un critère objectif qui permet de décider qu'une User Story a été traitée ou non*
- *Éviter que l'équipe ne commence trop de sujets sans réellement les finir*
- *Contribuer à la qualité de ce qui a été produit*

Exemple :

- *La revue de code a été effectuée*
- *Les tests d'acceptance sont validés par la QA*
- *Le Product Owner a vu la démo et a validé le fonctionnement*
- *Documentation réalisée, tests de charge validés, etc....*

TESTS D'ACCEPTANCE (ou critères d'acceptation)

Représentent un ensemble de condition que dois satisfaire une user story pour être considérée comme complète et terminée.

Rédigé par le PO, le PPO associé à l'équipe de développement

Formalisation suivant le langage Gherkin : GIVEN-WHEN-THEN

Exemple :

GIVEN : l'état du logiciel avant l'exécution de la story

WHEN : un évènement qui déclenche un processus

THEN : l'état du logiciel après exécution

Un test doit respecter la condition SMART :

SPECIFIC : compréhensible, facile à reproduire

MEASURABLE : quantifiable et observable

ACHIEVABLE : possible à réaliser (sans complexité excessive)

RELEVANT : approprié à la user story en question

TIME BOUND : avec un moment d'application circonscrit dans le temps

SPRINT (ou itération de développement)

Période de développement pendant laquelle l'équipe de développement s'engage à réaliser à un ensemble de fonctionnalités. Les périodes recommandées sont de deux ou trois semaines.

FACTEUR DE FOCALISATION

Capacité de production de l'équipe de développement sur une journée.

La valeur par défaut du facteur de focalisation est fixée à 0,7.

Cela signifie, qu'une ressource de développement produit de la valeur 70% de son temps. Les 30% restants correspondent à la gestion des mails, lectures des spécifications, pauses, etc.....

CAPACITE DE PRODUCTION

Nombre de points réalisable par une équipe de développement pendant un sprint,

*Règle de calcul : ((Nb jours calendaires - 1 jour cérémonies scrums) * Nb membres de l'équipe de développement) - nb jours d'indisponibilité des membres de l'équipes de développement (congrés, rtt, autres projets,...) * facteur de focalisation*

VERSION PRODUIT

Produit fonctionnel livré en fin de sprint

RELEASE

Ensemble de versions du produit corrigés, stabilisés et fonctionnelles. Généralement 1 release = 3 – 4 versions

METRIQUES (1/2)

VELOCITE

Nombre de points traité par l'équipe de développement pendant un sprint

FACTEUR DE FOCALISATION CONSTANTE

Bilan de la capacité de production de l'équipe de développement sur une journée sur un sprint.

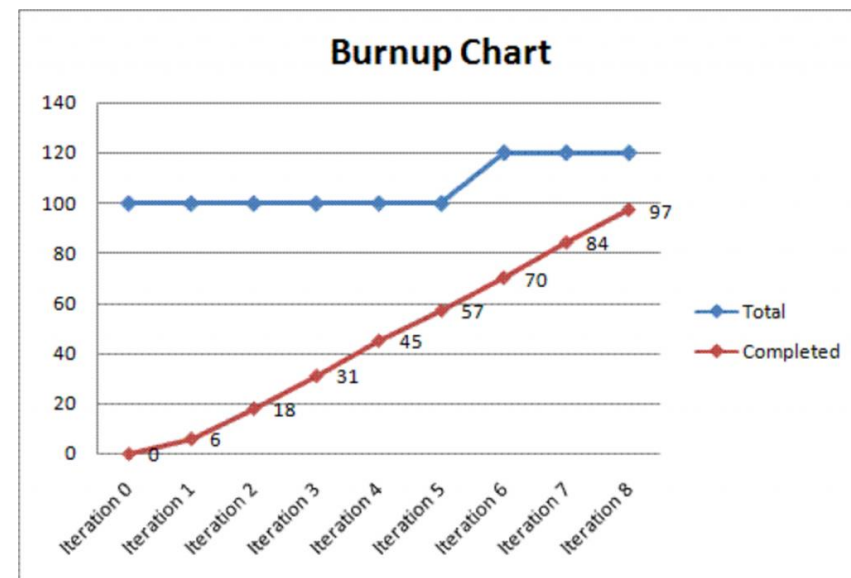
Calcul : VELOCITE / CAPACITE DE PRODUCTION



METRIQUES (2/2)

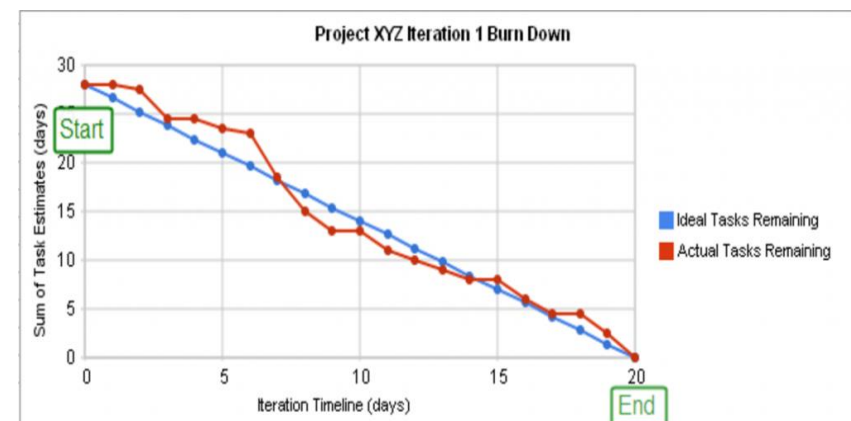
BURN-UP CHARTS

Graphique représentant l'évolution du produit (mesure la prise de valeur)



BURN-DOWN CHARTS

Graphique permettant de visualiser le reste à faire sur une période (sprint, version, release)



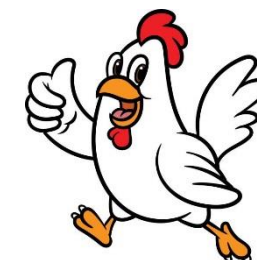
PIG (Cochon)

Désigne un acteur prenant part de manière active à une action ou à un rituel



CHICKEN (Poulet)

Désigne un acteur prenant étant spectateur d'une action ou à un rituel



III. LES ACTEURS

Mercredi 04 Novembre 2019



LA METHODE SCRUM S'ARTICULE AUTOUR DE 4 ACTEURS

PRODUCT OWNER

- Responsable pour **maximiser** la valeur du produit qui sera délivré
- **Priorise** les tâches du backlog
- **Arbitre** les points de blocage fonctionnel
- En charge du **suivi** globale du budget

PROXY PRODUCT OWNER

- **Second** le Product Owner dans la réalisation des différentes tâches

SCRUM MASTER

- **Garant** de la pratique et du respect de la méthodologie projet
- **Coach** de l'équipe de développement
- **Identifie** les risques / ou freins risquant d'impacter la bonne réalisation du produit
- **Limite le bruit** autour de l'équipe de développement
- Ne participe pas aux développements afin d'avoir une vision transverse du projet

EQUIPE DE DEVELOPPEMENT

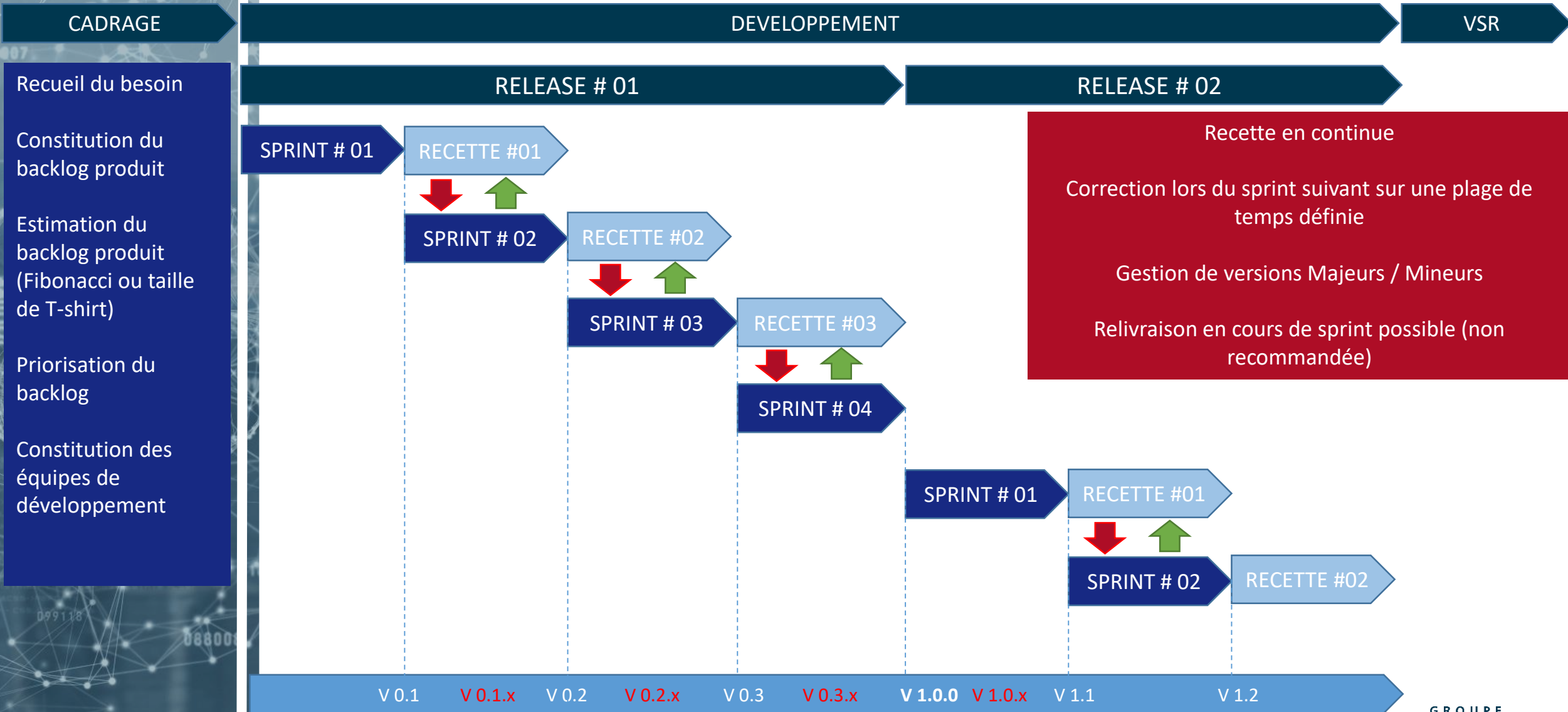
- Regroupe l'ensemble des intervenants **participants** au **développement** du produit :
 - Développeur back
 - Développeur front
 - Architecte applicatif
 - Designer
 - ...
- **Auto-organisée**
- Délivre une nouvelle **version fonctionnelle** du produit à chaque **itération** (et la plus **stable** possible)

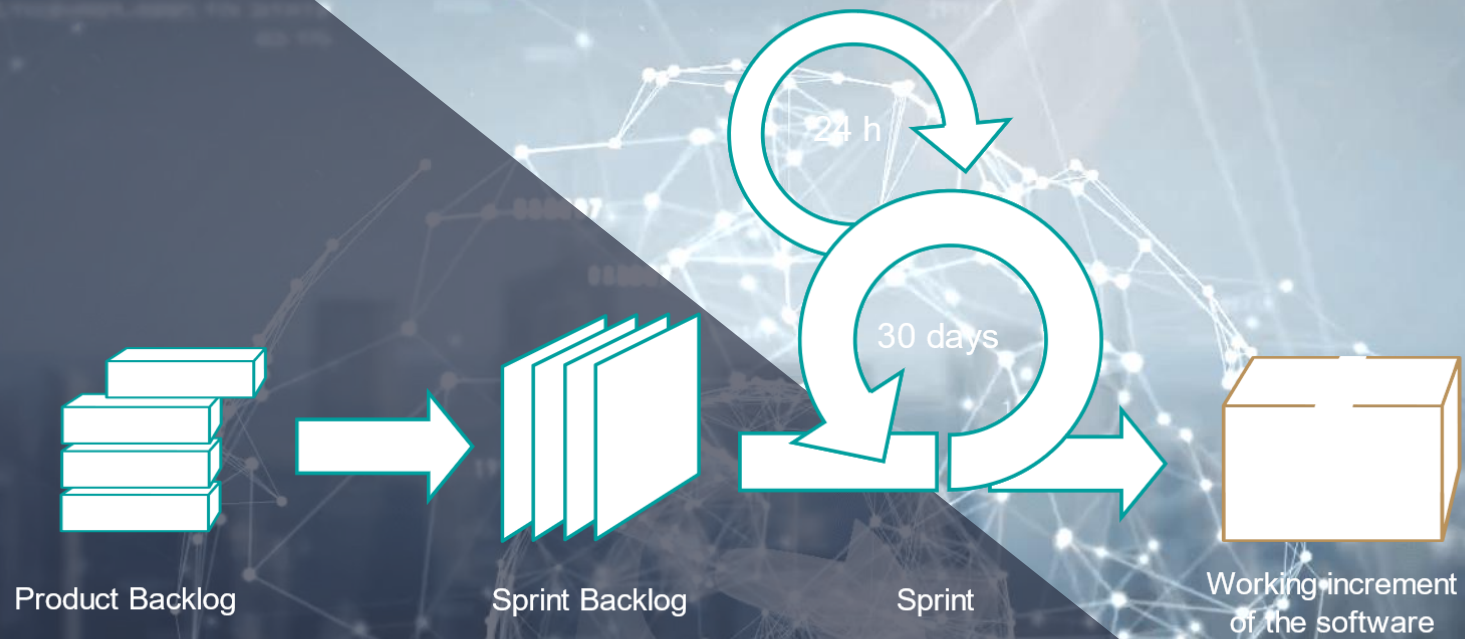
IV. ORGANISATION PROJET

Mercredi 04 Novembre 2019



ORGANISATION PROJET





V. LES RITUELS

Mercredi 04 Novembre 2019



PLANNING POKER



PARTICIPANTS

Product Owner / Proxy Product Owner

Scrum Master

Equipe de développement



DUREE

En fonction du backlog



FREQUENCE

1 / projet



OBJECTIFS

Chiffrage de l'ensemble du backlog

Chiffrage selon la suite de Fibonacci ou selon la taille de T-shirt



LIVRABLES

Product backlog chiffré



REMARQUES

Chiffrage revue à chaque sprint planning

SPRINT PLANNING



PARTICIPANTS

Product Owner / Proxy Product Owner

Scrum Master

Equipe de développement



DUREE

½ journée maximum



FREQUENCE

1 / sprint de développement



OBJECTIFS

Relecture des fonctionnalités à développer

Echange équipe de développement / Product Owner sur les fonctionnalités à développer afin d'éclaircir ou affiner le besoin

Révision du chiffrage des User-story en fonction des échanges DEV / PO ou PPO

Définir le contenu du sprint en fonction de la capacité de production

Prioriser les tâches à réaliser pendant le sprint

Définir le bonus du sprint (20% de la capacité de production prévisionnelle)



LIVRABLES

Product Backlog du sprint

Liste des objectifs du sprint



REMARQUES

Product backlog du sprint
=
Engagement de l'équipe de développement

DAILY MEETING


PARTICIPANTS

Product Owner / Proxy Product Owner
Scrum Master
Equipe de développement



DUREE

15 minutes maximum



FREQUENCE

1 / jour
Horaire fixe



OBJECTIFS

Echange sur les tâches / actions réalisées la veille

Organiser la journée de développement

Lister les douleurs (point de blocage, retard, risque, contraintes, ...) pouvant engendrer un risque

Lever d'éventuelles alertes

Mise à jour du tableau Scrum

Révision de la charge restante sur les tâches (+ ou -)



LIVRABLES

Mise à jour du tableau scrum
Mise à jour du Burn Down



REMARQUES

Les points à approfondir sont à réalisés
en dehors du daily meeting

SPRINT REVIEW (DEMONSTRATION)



PARTICIPANTS

Tous les acteurs du projets



DUREE

½ journée maximum



FREQUENCE

1 / sprint de développement



OBJECTIFS

Présenter le compte rendu du sprint

Présenter les fonctionnalités développées pendant le sprint en parcourant un scénario de démonstration basé sur les tests d'acceptance



LIVRABLES

Compte rendu du sprint

Version du produit fonctionnel déployée sur un environnement de recette

Livraison de la mise à jour de la documentation



REMARQUES

Les développeurs présentent les fonctionnalités

Séance interactive

SPRINT RETROSPECTIVE



PARTICIPANTS

Product Owner / Proxy Product Owner

Scrum Master

Equipe de développement



DUREE

2h maximum



FREQUENCE

1 / sprint de développement



OBJECTIFS

Séance de libre échange entre les différents interlocuteurs en charge de la réalisation du produit

Lister les points positifs et les axes d'amélioration du sprint

Pour chaque axe d'amélioration, définir des scénarios d'amélioration (principe d'amélioration continue)

Révision du bilan du sprint review du sprint précédent



LIVRABLES

Compte rendu de sprint review

Liste des actions à réaliser



REMARQUES

Aucun tabou

SPRINT REVIEW (DEMONSTRATION)



REMARQUES



La priorisation du sprint peut évoluer en cours de sprint



Toutes entrées dans le backlog du sprint engendre une sortie de fonctionnalités d'un poids équivalent



L'équipe s'engage sur les fonctionnalités à réaliser dans la sprint.

➔ Prise d'engagement sur le résultat



Le produit est fonctionnel à chaque version



Le produit est fonctionnel et stable à chaque release

Mercredi 04 Décembre 2019

