

L'agilité (SCRUM & KANBAN)

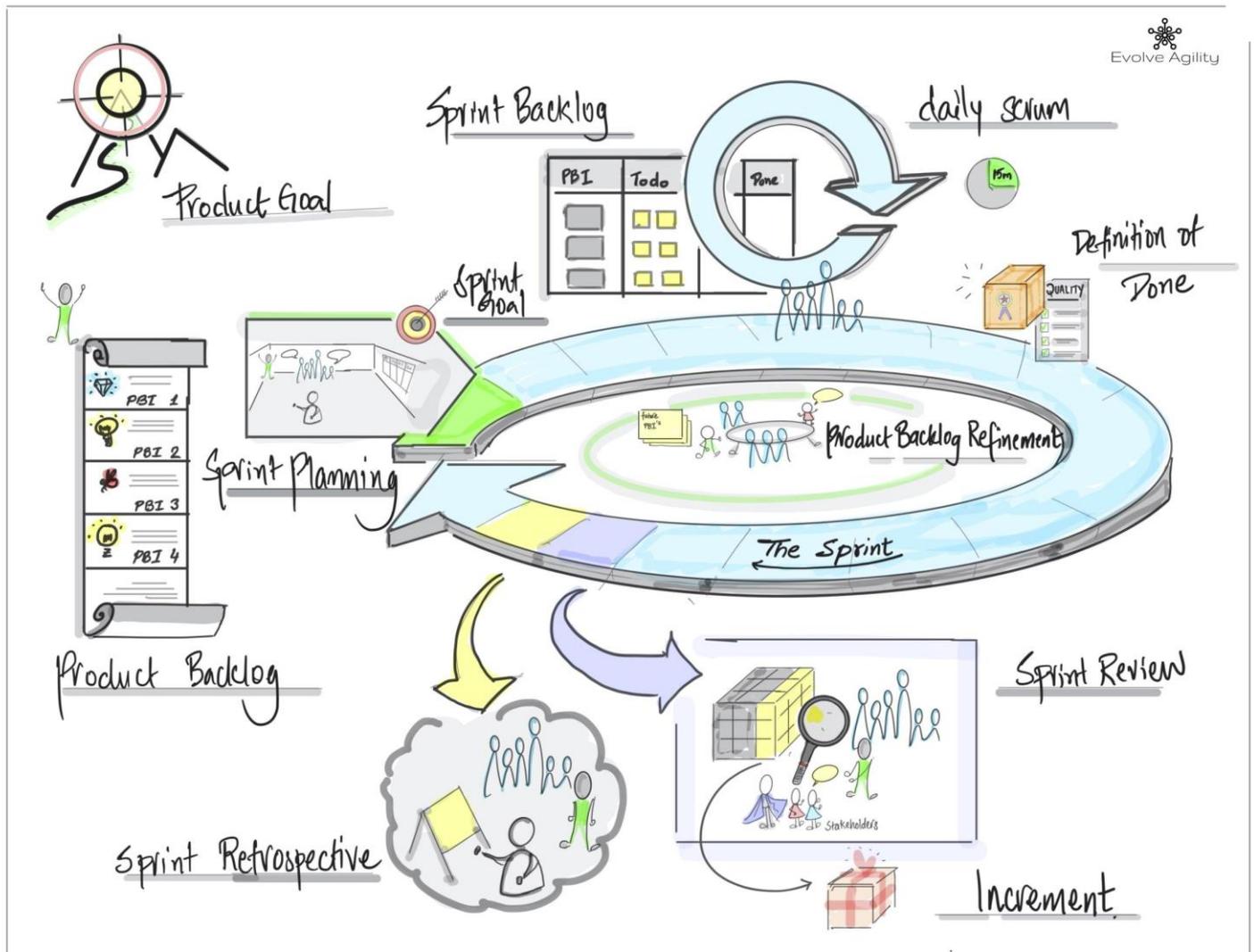


Table des matières

L'agilité (SCRUM & KANBAN)	1
Introduction à l'agilité	3
Les frameworks agile	3
SCRUM	4
Introduction	4
Les piliers de SCRUM	4
Les rôles, définitions et objectifs	5
Les évènements	8
Sprint planning	8
Daily SCRUM	9
Sprint review	9
Sprint rétrospective	9
Artéfacts du SCRUM.....	9
Kanban	10
Le tableau.....	10
Les règles du Kanban	11
Bibliographie	12

Introduction à l'agilité

L'agilité est un terme qui, dans le contexte du développement de logiciels et de la gestion de projets, se réfère à un groupe de méthodologies et de pratiques visant à répondre aux besoins clients de manière efficace tout en acceptant le changement comme une constante.

L'agilité a été définie de façon formelle avec la publication du Manifeste Agile en 2001, énonçant une série de valeurs et de principes favorisant une approche de développement logiciel itérative, incrémentale, et adaptable.

Les principes fondamentaux de l'agilité incluent la collaboration constante entre l'équipe de développement et les clients, la capacité de s'adapter aux changements même à un stade avancé du développement et le but de délivrer un produit fonctionnel à intervalles réguliers, assurant ainsi une satisfaction client continue et une orientation vers la qualité et l'efficacité.

L'agilité trouve son origine dans diverses méthodes de développement de logiciels qui ont émergé dans les années 1990 en réaction aux modèles plus traditionnels et rigides tels que le modèle en cascade. Elle représente une évolution de la gestion de projet qui met l'accent sur des cycles de développement courts et flexibles, la minimisation des risques par des livraisons fréquentes et l'amélioration continue à travers une auto-organisation et une réflexivité des équipes de travail.

L'approche agile s'est rapidement développée pour devenir l'un des cadres de travail de référence dans l'industrie du logiciel et s'étend désormais à d'autres domaines nécessitant une gestion de projet souple et adaptative.

Les frameworks agile

Les frameworks agiles sont diverses méthodologies qui appliquent les principes de l'agilité de manière systématique et structurée. Ils fournissent un squelette de pratiques et de conventions qui guident les équipes dans le développement de produits et la gestion de projets.

Chaque framework possède ses particularités, mais tous visent à réaliser des projets de manière incrémentale, à s'adapter rapidement aux changements et à mettre sur pied des équipes responsables et autonomes.

L'importance des frameworks agiles ne se limite pas à la production de résultats tangibles sous forme de produits ou services, mais englobe également l'amélioration de la communication interne, l'efficacité du travail d'équipe et l'adaptabilité des organisations face à des environnements en évolution rapide.

SCRUM

Introduction

SCRUM est un framework agile pour la gestion et la planification de projets de développement logiciel qui a vu le jour dans les premières années des méthodes agiles. Il a été conçu pour ajouter de la structure et de la discipline aux projets qui utilisent des méthodes agiles pour le développement de logiciels.

Ce cadre de travail s'appuie sur des itérations appelées Sprints, qui sont des cycles de travail de durée fixe durant lesquels une équipe vise à compléter et à livrer une partie fonctionnelle du produit final. Il est particulièrement réputé pour son solide cadre de rôles, de rituels et d'artefacts qui aident les équipes à se concentrer sur l'efficacité et à rester alignées sur les objectifs définis.

Ce framework a été introduit dans le développement logiciel dans les années 90 par Ken Schwaber et Jeff Sutherland, s'inspirant d'études précédentes sur des équipes hautement productives et des méthodes de fabrication lean.

Le terme "SCRUM" lui-même est emprunté au rugby, où il désigne une manière de relancer le jeu qui nécessite un travail d'équipe étroit et coordonné. Depuis sa création, SCRUM est devenu l'un des frameworks agiles les plus populaires en raison de sa flexibilité, de son organisation simple et de son approche pragmatique du travail d'équipe et de la gestion de projet.

Les piliers de SCRUM

SCRUM repose sur trois piliers fondamentaux : la transparence, l'inspection et l'adaptation.

1. La transparence assure que tous les aspects du travail sont visibles pour ceux qui en assument la responsabilité.
2. L'inspection permet aux membres de l'équipe de scruter le travail en cours sans affecter indûment leur productivité.
3. L'adaptation fait référence à la capacité d'ajuster le cours de l'action pour répondre aux changements. Cela implique des cycles de feedback récurrents qui assurent que le produit en développement reste aligné avec les besoins du client et les objectifs du projet.

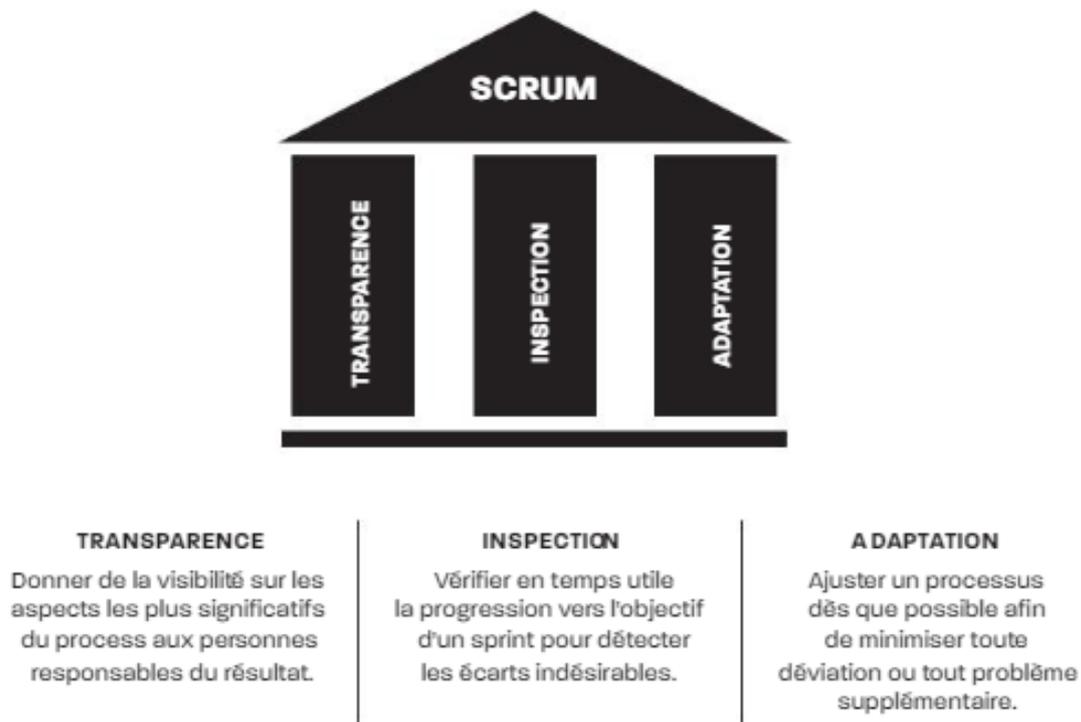


Figure 6.1. Les 3 piliers de Scrum

1

Les rôles, définitions et objectifs

Dans le SCRUM team, il existe trois rôles principaux : le Product Owner, le Scrum Master et l'Équipe de développement.

1. Le Product Owner est responsable de la valeur du produit final et travaille étroitement avec l'équipe pour identifier et prioriser les caractéristiques du produit.
 - Responsabilités :
 - Définit les fonctionnalités du produit en se basant sur les retours et les besoins des clients ou des parties prenantes.
 - Priorise le backlog de produit pour déterminer les tâches à effectuer en premier et celles pouvant attendre.
 - Assure la communication entre les parties prenantes et l'équipe de développement.
 - Accepte ou rejette les incréments de produit à la fin de chaque Sprint.
 - S'assure que la vision et les objectifs du produit sont clairs pour l'équipe de développement.
 - Tâches :

¹ Guide du product management (Venturino, Marcel) p. 151

- Maintient à jour la liste des fonctionnalités du produit, connue sous le nom de Product Backlog.
 - Planifie les Sprints en collaboration avec l'équipe de développement et le Scrum Master.
 - Participe aux différents événements Scrum (Planning, Review, Retrospectives, et le Daily Scrum en tant que spectateur si nécessaire).
 - Gère le budget, le calendrier et les attentes des parties prenantes.
2. Le Scrum Master agit comme un facilitateur pour l'équipe, aidant à éliminer les obstacles et à garantir que le processus SCRUM est suivi adéquatement.
- Responsabilités :
 - Assure que l'équipe Scrum comprend et utilise les valeurs, pratiques et règles de Scrum.
 - Aide l'équipe à devenir autonome, à maximiser sa productivité et à améliorer ses pratiques.
 - Travaille pour éliminer les obstacles qui pourraient entraver l'équipe de développement.
 - Protège l'équipe de développement des interférences extérieures pouvant perturber le processus de travail.
 - Facilite tous les événements Scrum nécessaires (Planning, Daily Scrum, Review, Retrospective).
 - Tâches :
 - Facilite la communication et la collaboration entre l'équipe de développement et le PO.
 - S'assure que les outils et les processus sont en place pour que l'équipe puisse travailler efficacement.
 - Coache l'équipe et l'organisation dans leur compréhension et pratique de Scrum et de l'agilité.
3. L'Équipe de développement comprend les professionnels qui travaillent ensemble pour livrer un incrément de produit potentiellement livrable à chaque Sprint.
- Responsabilités :
 - Construit le produit ou système en développant les fonctionnalités identifiées dans le Product Backlog.
 - Maintient la qualité du produit à travers des pratiques comme l'intégration continue, le TDD (Test-Driven Development) et le pair programming.
 - S'auto-organise pour organiser et répartir le travail sans dépendre directement de directives extérieures.
 - Participe à tous les événements Scrum nécessaires, notamment le Planning, le Daily Scrum, la Review et la Retrospective.
 - Communique clairement l'état d'avancement du travail et les défis rencontrés.
 - Tâches :

- Conçoit, code, teste et déploie les fonctionnalités du produit.
- Collabore étroitement les uns avec les autres pour assurer une cohérence dans le produit en cours de développement.
- Évalue les travaux à effectuer lors des sessions de planification et s'engage collectivement sur ce qui sera réalisé pendant le Sprint.
- Applique des pratiques d'ingénierie du logiciel pour garantir la qualité et la maintenabilité du code.

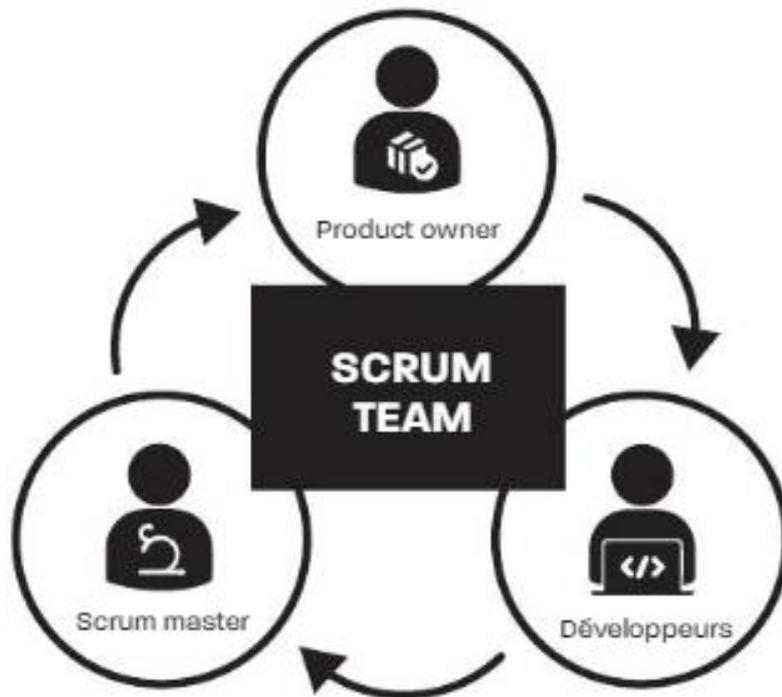


Figure 6.2. Composition de l'équipe Scrum

2

Ces rôles sont définis non pas pour créer une hiérarchie, mais pour distribuer les responsabilités et encourager l'efficacité et l'autogestion.

Chaque rôle dans une équipe Scrum est crucial et requiert une compréhension profonde et une adhésion aux principes de l'agilité pour travailler efficacement dans un environnement dynamique et changeant. La collaboration et la communication ouverte entre ces rôles sont essentielles pour le succès du projet IT.

² Guide du product management (Venturino, Marcel) p. 152

Les évènements

SCRUM est ponctué par plusieurs événements clés qui encadrent chaque Sprint. Le Sprint lui-même est un time-box de durée fixe, souvent de deux à quatre semaines, durant lequel le produit est développé.

- Le Sprint Planning est une réunion au début de chaque Sprint où l'équipe sélectionne le travail à réaliser au cours de ce Sprint.
- Le Daily Scrum est une réunion quotidienne rapide où l'équipe synchronise ses activités et planifie les travaux du jour.
- Le Sprint Review est une démonstration à la fin du Sprint où le travail accompli est présenté aux parties prenantes.
- ³La Sprint Rétrospective est le dernier événement d'un Sprint et permet à l'équipe de réfléchir sur le processus et de s'engager dans l'amélioration continue.

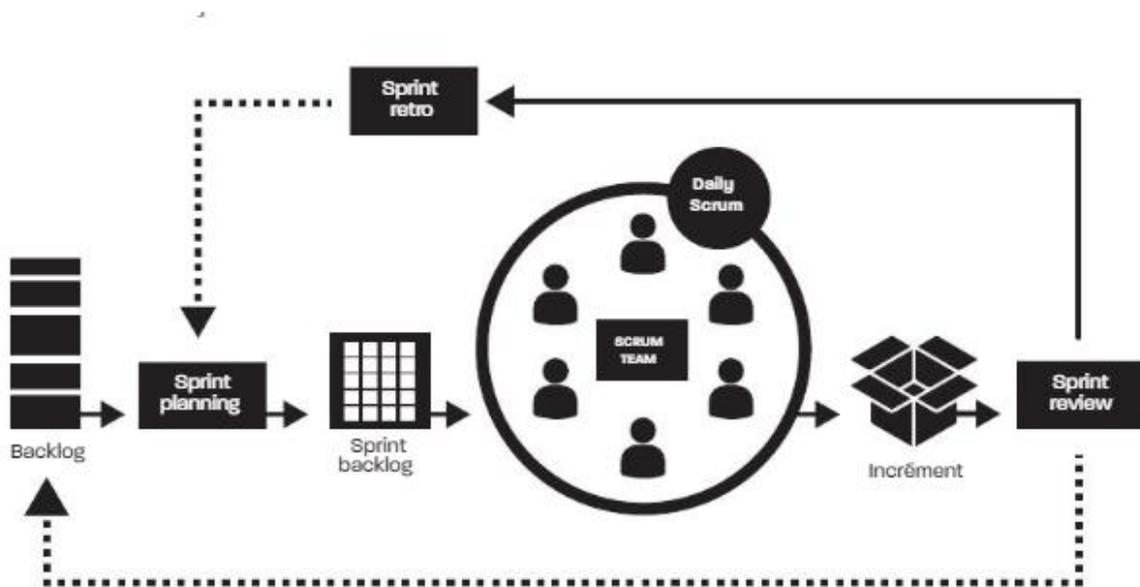


Figure 6.3. Représentation des événements du sprint selon le Scrum Guide

Sprint planning

Le Sprint Planning est une réunion qui fixe le ton pour le Sprint à venir. Durant cette session, l'équipe SCRUM, y compris le Product Owner et le Scrum Master, s'accorde sur les objectifs du Sprint et sur le backlog de Sprint, qui est un ensemble d'éléments choisis depuis le Product Backlog à implémenter pendant le Sprint.

La réunion de planification est essentielle car elle définit les attentes, identifie les potentiels défis et facilite un accord sur les priorités pour la période à venir. Pendant cette phase, le Product Owner

³ Guide du product management (Venturino, Marcel) p. 154

joue un rôle critique en clarifiant les détails des éléments du backlog et en aidant l'équipe à comprendre l'impact du travail prévu.

Daily SCRUM

Le Daily SCRUM, aussi appelé stand-up, est une réunion quotidienne de 15 minutes pendant laquelle l'équipe discute de l'avancement par rapport aux objectifs du Sprint. L'objectif est de promouvoir la communication rapide et efficace, de s'assurer que tous sont synchronisés et d'identifier de manière proactive les obstacles qui pourraient entraver la progression.

Il est crucial d'éviter les erreurs communes telles que le report de cette réunion, la transformation en séance de résolution de problèmes, ou la non-participation de membres de l'équipe. Pour une réunion quotidienne réussie, chaque membre de l'équipe devrait brièvement partager ce qu'ils ont fait la veille, ce qu'ils prévoient de faire aujourd'hui et les obstacles éventuels.

Sprint review

À la fin de chaque Sprint, l'équipe organise un Sprint Review pour présenter ce qu'elle a accompli. Cette revue est une opportunité pour l'équipe de démontrer les nouvelles fonctionnalités et pour les parties prenantes de fournir des retours en temps réel qui orienteront les prochains cycles de travail.

L'évaluation de la revue du Sprint consiste à confirmer que les objectifs du Sprint ont été atteints et à discuter des éventuelles améliorations ou ajustements. Les éléments clés de cette réunion incluent la démonstration des fonctionnalités développées, l'évaluation de l'alignement sur les objectifs du produit et la discussion sur ce que l'équipe devrait entreprendre ensuite.

Sprint rétrospective

La rétrospective est une réunion de réflexion qui se déroule après le Sprint Review et avant le prochain Sprint Planning. Son but est de permettre à l'équipe de discuter ouvertement de ce qui a bien fonctionné et de ce qui pourrait être amélioré.

Pendant la rétrospective, l'équipe identifie des stratégies concrètes pour résoudre les problèmes et développer une approche plus efficace pour le prochain Sprint. L'importance de cette phase ne se limite pas au développement du produit, mais englobe aussi le bien-être de l'équipe et le renforcement de sa cohésion.

Artéfacts du SCRUM

SCRUM utilise une série d'artéfacts pour capter et partager les informations clés nécessaires aux décisions au cours d'un projet.

- Le Product Backlog est une liste ordonnée contenant tout ce qui pourrait être nécessaire dans le produit et constitue la source unique des exigences pour tout changement à apporter au produit.

- Le Sprint Backlog est un ensemble d'éléments du Product Backlog sélectionnés pour le Sprint, ainsi que le plan pour livrer ces éléments et atteindre l'objectif du Sprint.
- Les User Stories sont des descriptions courtes, simples et compréhensibles des fonctionnalités à développer, écrites du point de vue de l'utilisateur final, et les story points sont une unité de mesure utilisée pour estimer la difficulté de ces histoires.

Ces artefacts sont essentiels pour maintenir la transparence et assurer la cohérence des attentes entre l'équipe et les parties prenantes.

Kanban

Le tableau

Le Kanban est un autre framework agile qui s'appuie sur la visualisation du flux de travail ainsi que sur la gestion et la limitation des travaux en cours. Le tableau Kanban est l'élément central de cette méthode et sert à visualiser le travail pour chaque étape du processus de développement.

Généralement, un tableau Kanban est partitionné en colonnes qui représentent différentes phases de travail, telles que "À faire", "En cours" et "Terminé". Les tâches sont représentées par des cartes qui glissent d'une colonne à l'autre, illustrant ainsi leur progression vers l'achèvement.

L'utilisation d'un tableau Kanban permet une identification facile des blocages et des surcharges, et encourage l'efficacité par l'amélioration continue du processus de travail.

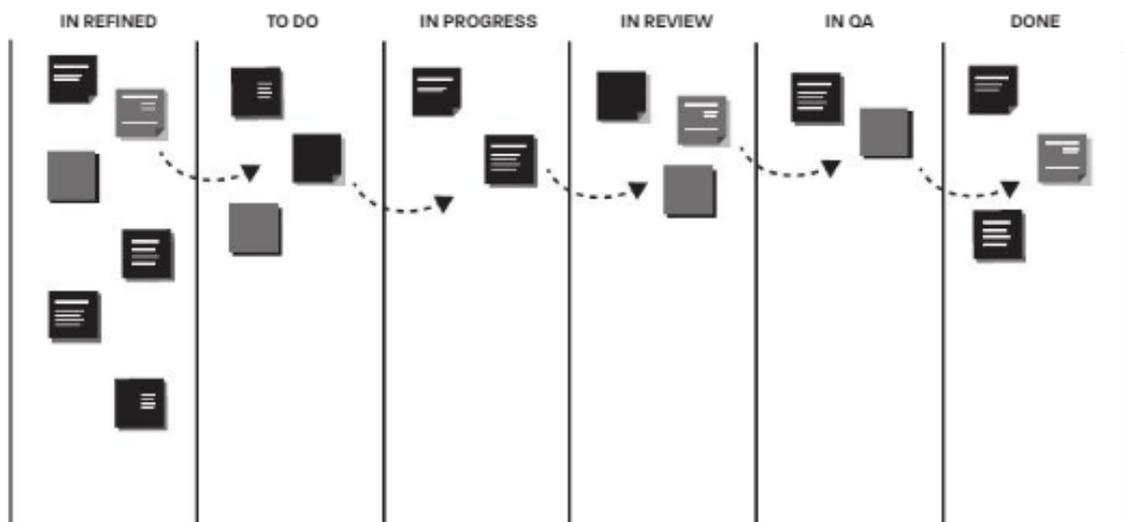


Figure 6.4. Représentation d'un tableau Kanban, avec ses différentes colonnes

Les règles du Kanban

Kanban repose sur quatre principes fondamentaux : visualiser le travail, limiter les travaux en cours (WIP), gérer le flux et s'améliorer continuellement.

1. La visualisation du travail par un tableau Kanban aide à comprendre la progression du projet et à identifier les problèmes éventuels.
2. La limitation des travaux en cours permet d'accroître la focalisation et de réduire le temps de cycle, favorisant ainsi la finalisation rapide des tâches.
3. La gestion du flux est axée sur l'amélioration continue du processus de travail et du débit.
4. Enfin, l'amélioration continue est obtenue par des ajustements constants pour optimiser l'efficacité et la qualité du travail produit.

Kanban est donc particulièrement apprécié pour sa simplicité, son adaptabilité et son potentiel de renforcer la collaboration au sein des équipes.

Bibliographie

Businessnews Publishing. (2016). *Summary: Scrum*. Business Book Summaries.

Venturino, M. (2023). *Guide du product management*. Eyrolles.