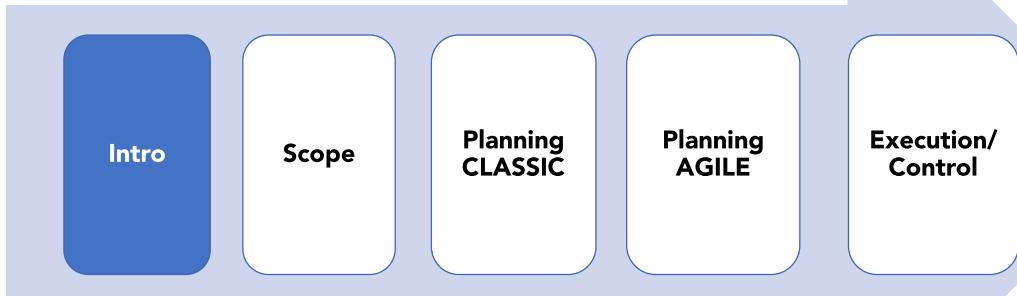


ING02 – PM Refresh



Sondage....

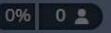
Allez sur wooclap.com et utilisez le code **GMAQIR** 

Lors de votre année d'ING1, comment avez vous suivi la formation en Gestion de Projet ?

 **1** J'ai suivi tous les cours en amphi et j'ai participé au deux TDs  0 / 0 

 **2** Je suis allé seulement en TD
Cliquez sur l'écran projeté pour lancer la question  0 / 0 

 **3** Je ne suis allé ni au cours en amphi, ni aux TDs, et j'ai révisé le partieil en utilisant les supports de cours.  0 / 0 

wooclap   105 %   0 / 0 

Sondage suite...

Allez sur wooclap.com et utilisez le code **GMAQIR**

Lors de votre Stage de deuxième année, avez-vous

- ① Participé à un projet ? 0% | 0 people
- ② Effectué des développements en mode AGILE ? 0% | 0 people
Cliquez sur l'écran projeté pour lancer la question
- ③ Utilisé des éléments enseignés dans le cours de gestion de projet ? 0% | 0 people

wooclap 105 % 0 / 0 people

Un projet est une entreprise temporaire initiée dans le but de fournir un produit, un service ou un résultat unique.

La nature **temporaire** des projets indique une date de début et une date de fin **définies**. La fin est atteinte lorsque les objectifs du projet ont été réalisés, ou lorsque le projet est terminé parce que ses objectifs ne seront pas ou ne peuvent pas être atteints, ou lorsque le besoin du projet n'existe plus.

Les projets existent pour créer un produit ou un service **qui n'existait pas auparavant**. En ce sens, un projet **est unique**. Unique signifie que c'est nouveau, et que cela n'a jamais été fait auparavant. Cela a peut-être déjà été fait de manière très similaire, mais jamais exactement de cette manière.

Un projet est entrepris pour atteindre des objectifs en produisant des **éléments livrables**.

Les projets produisent un changement : d'un état A vers un état B
Les projets permettent la création de valeur commerciale.

Un projet possède des attributs distinctifs, qui le distinguent du travail en cours, ou des opérations commerciales. Les projets sont temporaires par nature. Ils ne font pas partie des processus opérationnels quotidiens et ont **des dates de début et de fin définitives**.

Cette caractéristique est importante car une grande partie des efforts déployés dans le cadre d'un projet sont consacrés à garantir que le projet sera achevé à la date prévue. Pour ce faire, des calendriers sont créés, indiquant quand les tâches doivent commencer et se terminer. Les projets peuvent durer quelques minutes, quelques heures, quelques jours, quelques semaines, quelques mois ou quelques années.

La Gestion de Projet :

- Une approche **pragmatique**
- Un ensemble **de techniques et de processus**
dont l'objectif et de maximiser les chances de succès du projet
- Il convient de les **adapter** au type de projet

Le PM (Project Manager) va utiliser un ensemble de techniques et de Best Practices qui vont l'aider à structurer le projet et à piloter son exécution,
Le but et les bénéfices de ces méthodes seront de faire arriver le projet à temps, d'anticiper les dérives: **de maximiser la réussite du projet.**

Suivant la taille et la nature du projet, on utilisera seulement une partie de ces techniques.

Les 10 domaines de connaissance

- **Gestion de l'intégration**
- **Gestion du périmètre du projet**
- **Gestion du calendrier**
- **Gestion des coûts du projet**
- **Gestion de la qualité du projet**
- **Gestion des ressources**
- **Gestion des risques**
- **Gestion des communications du projet**
- **Gestion des achats du projet**
- **Gestion des parties prenantes**

Allez sur wooclap.com et utilisez le code **GMAQIR** 

Vous démarrez un nouveau Projet, par exemple de Fin d' Etude Quel est le premier élément à établir ?  

 1 Le Product Backlog

0% 

  2 Le Project Charter



0% 

Cliquez sur l'écran projeté pour lancer la question

 3 La Vision Produit

0% 

 4 Les Spécifications Fonctionnelles

0% 

wooclap

🔒 🔎 105 % 🔎

0 / 0 



La Charte Projet



Au lieu d'une vague description des objectifs du projet, **la Charte Projet** (Project Charter), aussi appelée note de cadrage, est un document synthétique qui contient l'ensemble des caractéristiques qui décrivent le projet.

Son contenu est le suivant :

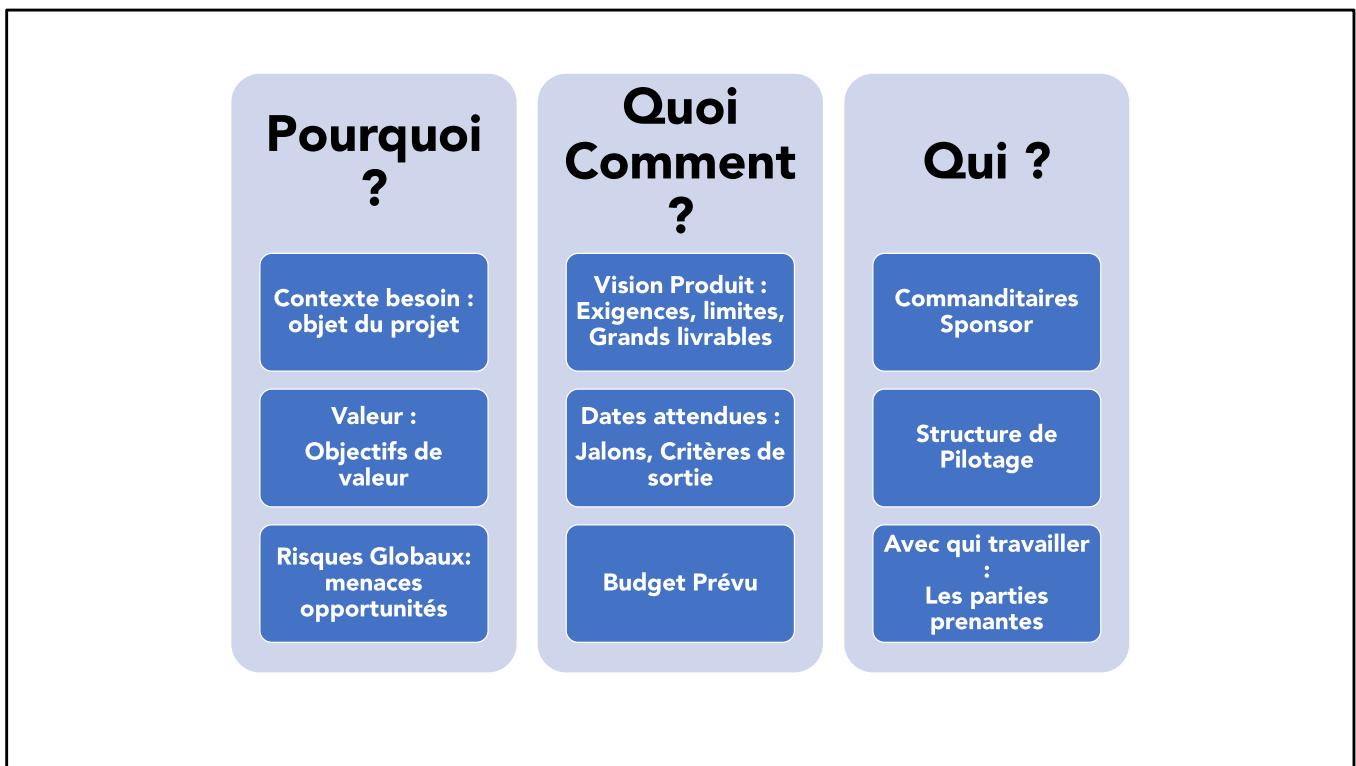
- * Objet du projet.
- * **Objectifs mesurables** du projet, et les

critères de succès.

- * Grandes lignes des **exigences**.
- * Grandes lignes du projet, de ses limites et **les livrables** principaux.
- * **Risques Globaux**.
- * Résumé de l'échéancier et des **jalons** principaux.
- * **Ressources financières** préapprouvées, disponibles pour le projet.
- * Listes des **parties prenantes** principales.
- * Circuits d'approbation des décisions.
- * Critères de sortie du projet de ses phases principales.
- * Nom du Chef de projet assigné, ses responsabilités et son niveau d'autorité.

Equivalences Anglaises: PROJECT CHARTER; Project Purpose; Measurable project objectives and related success criteria; High-level requirements; High-level project description, boundaries, and key deliverables; Summary milestone schedule; Key stakeholder list; Overall project risk; Project approval

requirements.



Un objectif flou, et vous ne savez pas ce que l'on attend de vous ?

Vous risquez de faire un produit décentré/ attentes.

Objective is unclear. You don't know what is expected from your work ?

**Le client vous pilote sans
vous donner une vision
claire du projet.**

Customer is driving, but does not provide a clear vision of the project.

**Vous avez le Droit et le Devoir
de collecter les informations
de compréhension du
PROJECT CHARTER afin
d'établir une Vision Produit**

You can, and you **MUST** collect informations to build the **PROJECT CHARTER**

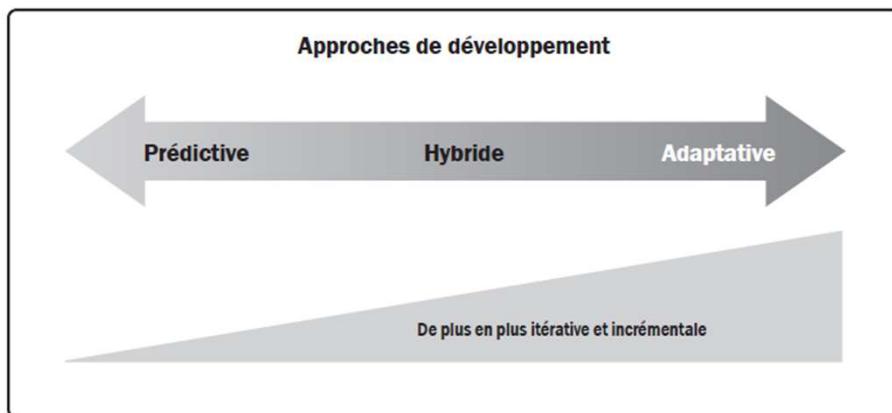
Exemple de Project Charter

MTI/PFEE



Choisir une démarche de mise en œuvre du Projet.

Approches de développement



Approche prédictive.

L'approche prédictive est utilisée lorsque les exigences du projet et du produit peuvent être définies, recueillies et analysées au début du projet. Elle est également appelée approche en cascade (waterfall). Cette approche est aussi utilisée dans le cas d'un investissement important et d'un niveau de risque élevé pouvant nécessiter les revues fréquentes, des mécanismes de maîtrise des changements et une re planification entre phases de développement. Le périmètre, l'échéancier virgule le coût, les ressources nécessaires et les risques peuvent être bien définies dans les premières phases du cycle de vie du projet. En outre ils sont relativement stables. Cette approche de développement permet à l'équipe projet de réduire le degré d'incertitude au début du projet et de planifier un maximum en amont. L'approche prédictive peut se fonder sur un travail de validation de principes pour étudier les possibilités, mais la majorité du travail du projet suit les plans établis au début du projet. Bien souvent, les projets qui utilisent cette approche disposent des modèles d'anciens projets similaires.

Approche adaptative.

L'approche adaptative est utile lorsque le degré d'incertitude et de volatilité des exigences est élevé, et que ces dernières sont susceptibles de changer au cours du projet. Une vision claire est établie au début du projet , et les exigences initiales sont connues précisées détaillées modifiées ou remplacées en fonction du retour d'information des utilisateurs et de l'environnement ou des événements inattendus.

L'approche adaptative fait appel aux approches itératives et incrémentales. Cependant à l'extrême des techniques adaptatives, ces itérations tendent à diminuer et le produit est plus susceptible d'évoluer en fonction du retour des informations des parties prenantes.

Approche hybride.

Les approches de développement hybrides combinent les approches adaptatives et prédictives. Cela signifie qu'elle utilise certains éléments de l'approche prédictive et d'autres de l'approche adaptative. Cette approche de développement est utile lorsque les exigences génèrent de l'incertitude ou du risque. L'approche hybride est également utile lorsque livrables peuvent être décomposés en modules, ou que certains peuvent être élaborés par différentes équipes du projet. Elle est plus adaptable que l'approche prédictive mais moins qu'une approche purement adaptative. Par exemple , un projet avec 2 livrables principaux dont l'un est développé à l'aide d'une approche adaptative et l'autre à l'aide d'une approche prédictive.

Allez sur wooclap.com et utilisez le code **GMAQIR** 

Pour quels types de projet choisir une approche Prédictive ??

 1 Proof Of Concept (POC)

0% | 0 



 2 Création d'une APP MOBILE interactive

0% | 0 



 3 Déploiement d'une infrast **Cliquez sur l'écran projeté pour lancer la question**

0% | 0 

 4 Ajout d'un ensemble de fonctions à une appli existante

0% | 0 

 5 Réalisation d'une appli technique contrainte en terme de dates

0% | 0 

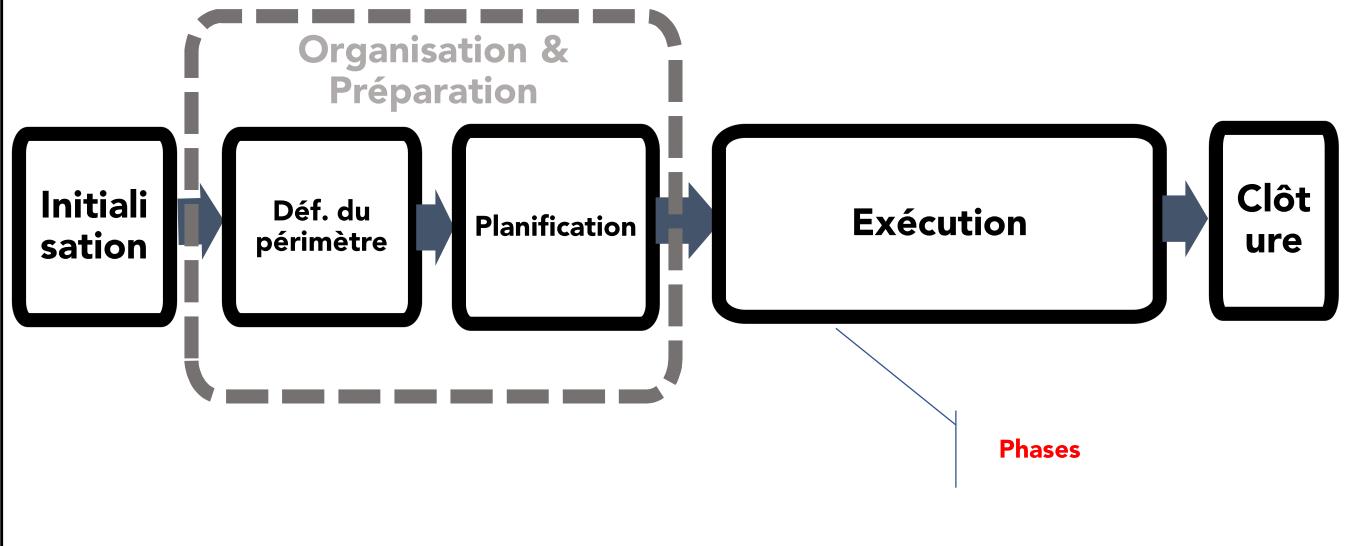
wooclap

🔒 🔍 105 % 🔍

0 / 0 



Cycle de vie Projet Prédictif:



Pour simplifier, nous découperons en 5 phases distinctes les activités de la gestion de projet:

- La Phase d'initialisation

La Phase Organisation et Préparation que l'on peut découper en

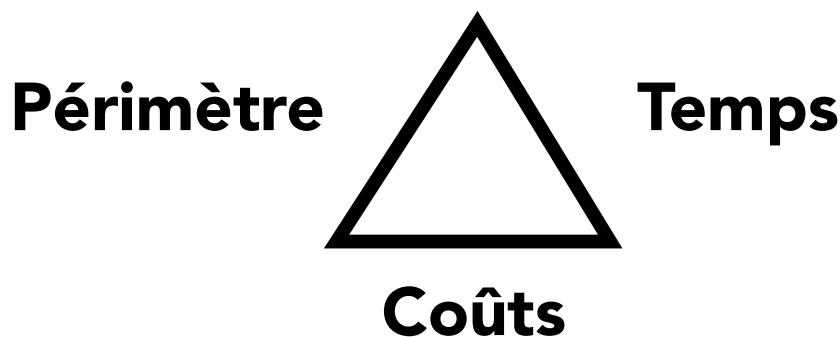
- Définition du contenu du projet
- La Planification
- La Phase d'exécution
- La Phase de Clôture

Le PMBOK de l'organisme PMI regroupe les activités de la gestion de projets en **processus** qui sont répartis en plusieurs **groupes**:

- 1) Initialisation
- 2) Planification
- 3) Exécution
- 4) Maitrise

5) Clôture

Logique d'un Projet Predictif :



Les trois domaines essentiels de la gestion de projet traitent des 3 dimensions essentielles d'un projet :

1) Tout d'abord le Périmètre (ou SCOPE) : Le périmètre défini **le QUOI ?** c'est-à-dire les fonctionnalités qui sont attendues du projet et ce que seront ses « livrables ». Il permet aussi d'établir clairement **ce qui est exclu du projet.**

2) Le Temps (Calendrier, Echéancier ou Schedule) qui sera alloué à l'exécution de ce produit. Ce planning temporel répond à la question **QUAND** ? seront fournis les livrables du projet, et donc quand il doit se terminer. Il est composé par les principaux **jalons (milestones)** du projet.

3) Les coûts, c'est-à-dire le budget qui sera alloué au projet. Combien de ressources humaines et matérielles seront affectées à ce projet. **COMBIEN** ? l'entreprise a-t-elle prévu de dépenser pour obtenir le résultat ou le produit que constitue ce projet.

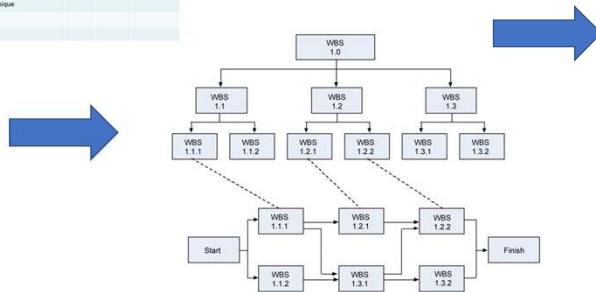
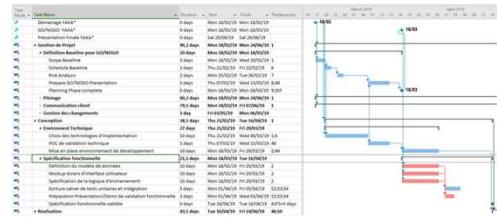
Le Chef de projet doit en permanence travailler sur l'équilibre entre ces 3 dimensions interdépendantes : rajouter une fonctionnalité dans le périmètre a probablement un impact sur la durée du projet et sur ses coûts...

Equivalences Anglaises: SCOPE, TIME COSTS , SCHEDULE , MILESTONE

CLASSIC/PREDICTIVE



#	Type	Description	Etat	Ecran Module	Commentaires
F001	Fonctionnel	Un utilisateur peut demander la création d'une association	A	SC00X	
F002	Fonctionnel	L'interface doit proposer un accès au planning de réservation des salles	C	SC00Y	
F003	Fonctionnel	Une fonction de création automatique d'une association existe.	R		
.....					
T001	Technique	La connexion au site de tous les utilisateurs doit se faire au travers du protocole HTTPS	A		
.....					
C001	Contrainte	Respect de la carte graphique			
.....					



With CLASSIC PREDICTIVE APPROACH:

- 1) We are confident that we know what should be the results of the project
- 2) That Requirements will vary only marginally
- 3) Every single task from beginning to end of the project is scheduled
- 4) There is a high level of understanding on how work will be performed

CLASSIC PREDICTIVE approach comes from BUILDING and MANUFACTURING

It shows a working product at 2/3 of the way

CLASSIC PREDICTIVE may include a Prototype or Pilot phase to secure requirement

Avec l'approche CLASSIQUE ou PREDICTIVE :

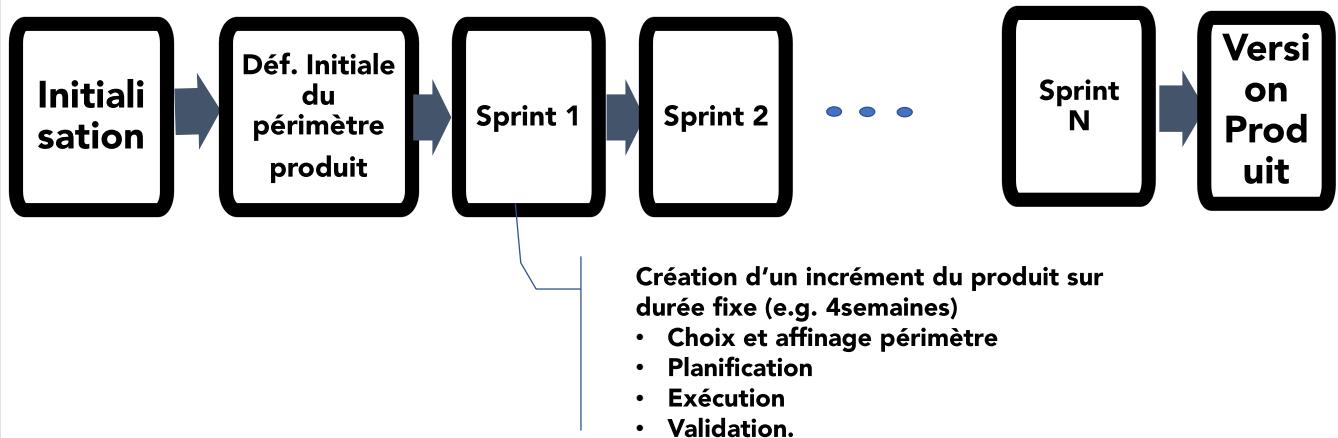
- 1) Nous sommes persuadés que nous savons quels devraient être les résultats du projet
- 2) Que les exigences ne varieront que marginalement
- 3) Chaque tâche du début à la fin du projet est prévue
- 4) Nous avons un haut niveau de compréhension de la façon dont le travail sera effectué

L'approche PREDICTIVE ou CLASSIQUE vient des métiers de la Construction et du Manufacturing

On est capable de présenter un produit au 2/3 du temps complet du projet

Cette Methode n'interdit pas d'inclure une phase prototype ou pilote pour confirmer les exigences.

Cycle de vie Projet AGILE:



La Méthode de gestion de Projet « Classique » est désigné sous le vocable de PREDICTIVE.

C'est-à-dire que dans la phase de préparation on définit le périmètre de tout le projet et on planifie l'ensemble des tâches pour réaliser le projet.

La Méthode AGILE est une méthode plus empirique , elle permet de construire le résultat par itération successives ou SPRINT d'une durée fixe relativement courte de 2 à 6 semaines. Chaque Itération produit un incrément et l'intégration des incréments permet de réaliser la version du produit attendue.

Logique d'un Projet Agile :

Priorité au produit : La valeur qu'il apporte et l'adéquation aux attentes.

Moins focalisation : Planning /budget

Les trois domaines essentiels de la gestion de projet traitent des 3 dimensions essentielles d'un projet :

1) Tout d'abord le Périmètre (ou SCOPE) : Le périmètre défini **le QUOI ?** c'est-à-dire les fonctionnalités qui sont attendues du projet et ce que seront ses « livrables ». Il permet aussi d'établir clairement **ce qui est exclu du projet.**

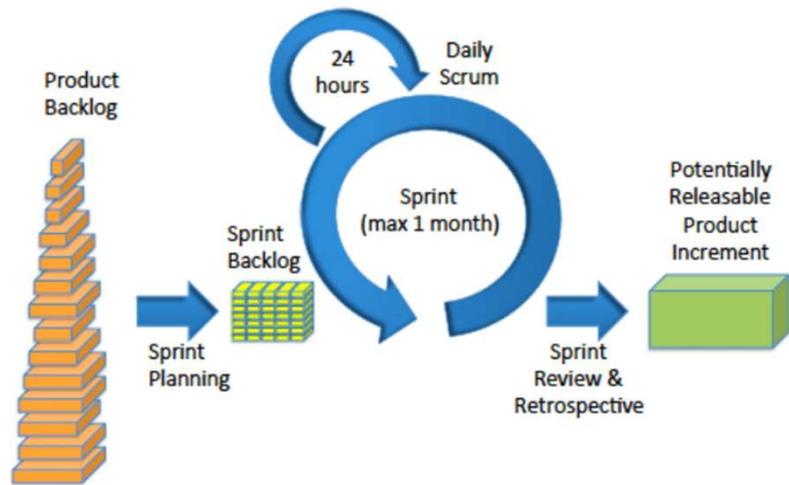
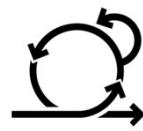
2) Le Temps (Calendrier, Echéancier ou Schedule) qui sera alloué à l'exécution de ce produit. Ce planning temporel répond à la question **QUAND** ? seront fournis les livrables du projet, et donc quand il doit se terminer. Il est composé par les principaux **jalons (milestones)** du projet.

3) Les coûts, c'est-à-dire le budget qui sera alloué au projet. Combien de ressources humaines et matérielles seront affectées à ce projet. **COMBIEN** ? l'entreprise a-t-elle prévu de dépenser pour obtenir le résultat ou le produit que constitue ce projet.

Le Chef de projet doit en permanence travailler sur l'équilibre entre ces 3 dimensions interdépendantes : rajouter une fonctionnalité dans le périmètre a probablement un impact sur la durée du projet et sur ses coûts...

Equivalences Anglaises: SCOPE, TIME COSTS , SCHEDULE , MILESTONE

AGILE: SCRUM



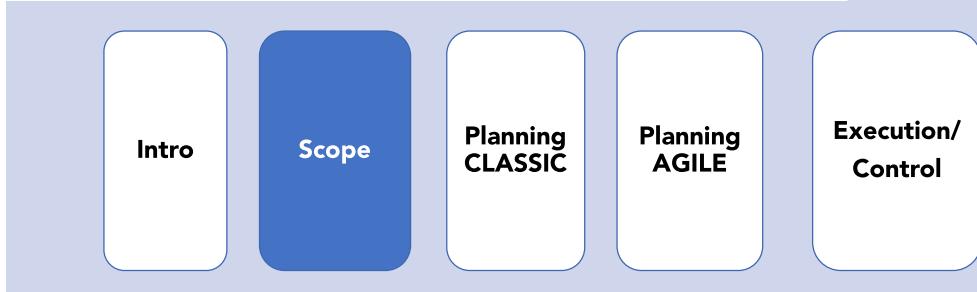
AGILE : Approach based on SCRUM: <https://www.scrumguides.org/index.html>

En résumé :

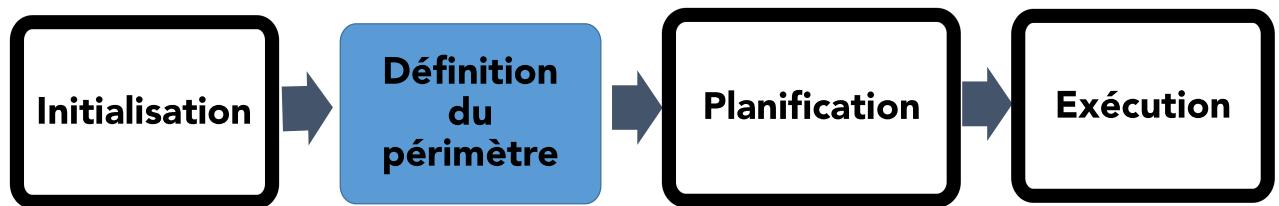
En fin de phase initiale, vous savez :

- **Pourquoi le projet est lancé,**
- **Quel résultat est attendu,**
- **Dans quel cadre le projet doit se dérouler,**
- **Les dates clefs attendues,**
- **Quels seront vos interlocuteurs.**

ING02 – PM Refresh



Cycle de vie Projet:



Le terme **Périmètre** désigne le contenu précis du produit qui sera le résultat du projet :

Le Contenu ou Périmètre produit, ce sont les caractéristiques et les fonctions qui caractérisent un produit, un service ou un résultat. Ce contenu est, tout d'abord décrit en termes d'exigences qui décrivent précisément les fonctions attendues du produit.

Dans le mode prédictif, on va construire un **Tableau d'exigences** pour classifier et regrouper toutes les attentes liées au produit,

Dans le mode itératif et plus spécialement dans la méthode AGILE, on formalisera ces attentes dans un **PRODUCT BACKLOG**.

Equivalences Anglaises : PRODUCT SCOPE.
REQUIREMENT MATRIX

Le Périmètre Produit



**Décrire le produit -> Référentiel contenant
Exigences Fonctionnelles & Contraintes**



Exigences ≠ Specifications



Le terme **Périmètre** désigne le contenu précis du **Produit** qui sera le résultat du projet:

Le **Contenu ou Périmètre produit**, ce sont les caractéristiques et les fonctions qui caractérisent un produit, un service ou un résultat. Ce contenu est, tout d'abord décrit en termes d'exigences qui décrivent précisément les fonctions attendues du produit. Il s'agit donc bien de formuler le besoins des utilisateurs , du clients ...

Dans le cas d'un projet de développement d'application, le périmètre est défini par un **ensemble d'exigences fonctionnelles** qui décrivent les attentes par rapport au produit logiciel qui doit être livré. Ce sont les fonctions du logiciel qui sera produit comme résultat du projet.

Ces exigences sont complétées par un ensemble de contraintes:

On distinguera les contraintes d'ordre technique, comme l'environnement cible qui devra supporter l'application, d'ordre projet comme les contraintes de temps, de ressources ou d'interface, et les contraintes qualité comme le respect de certaines contraintes réglementaires, le niveau de performances attendues, la compatibilité avec d'autres logiciels ou la conformité à certains standards.

Il faudra, pour disposer d'une description complète de ces exigences:

- 1) Les recueillir – à travers diverses techniques
- 2) Les formaliser, c'est-à-dire les classifier et les prioriser

Le terme **Périmètre** désigne le contenu précis du **Produit** qui sera le résultat du projet :

Le **Contenu ou Périmètre produit**, ce sont les caractéristiques et les fonctions qui caractérisent un produit, un service ou un résultat. Ce contenu est, tout d'abord décrit en termes d'exigences qui décrivent précisément les fonctions attendues du produit

Le **périmètre du Projet**, c'est l'ensemble du travail qui doit être réalisé pour livrer le Produit.

Dans le cas d'un projet de développement d'application, le périmètre est défini par un **ensemble d'exigences fonctionnelles** qui décrivent les attentes par rapport au produit logiciel qui doit être livré. Ce sont les fonctions du logiciel qui sera produit comme résultat du projet.

Ces exigences sont complétées par un ensemble de contraintes :

On distinguera les contraintes d'ordre technique, comme l'environnement cible qui devra supporter l'application, d'ordre projet comme les contraintes de temps, de ressources ou d'interface, et les contraintes qualité comme le respect de certaines contraintes réglementaires, le niveau de performances attendues, la compatibilité avec d'autres logiciels ou la conformité à certains standards.

Il faudra, pour disposer d'une description complète de ces exigences :

- * Les recueillir – à travers diverses techniques
- * Les formaliser, c'est-à-dire les classifier et les prioriser

Attention : Les exigences (ou requirements) ne sont pas des spécifications.

Il faut bien faire la différence entre **les attentes du client** par rapport au produit et **la solution proposée** : les exigences décrivent les attentes et les spécifications décrivent la solution proposée par le projet.

Dans le cas d'un appel d'offres, il y aura probablement plusieurs solutions proposées à partir du même jeu d'exigences fonctionnelles contenues dans le cahier des charges.

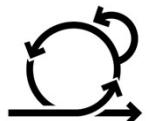
Equivalences Anglaises : FUNCTIONAL REQUIREMENTS

Exigence fonctionnelle :

Ce que le client attend comme fonction de notre produit.

Exemples pour une application :

- Permet de créer un compte
- Permet de rejoindre un groupe de discussion
- Permet de prendre un RDV pour se faire Vacciner
- Permet de publier une recette de cuisine visible par son groupe



User Story:

**En tant que <qui>, je veux
<quoi> afin de <pourquoi>.**

L'intégralité du travail à réaliser est découpée en incrément fonctionnels et les activités de développement s'organisent autour de ces incrément appelés "User Stories".

Techniquement il va s'agir de "morcelez" l'ensemble des objectifs fonctionnels d'un projet en User Stories.

Adopter la pratique des User Stories implique de tenir pour généralement vraies un certain nombre d'hypothèses sur ces incrément: on peut les réaliser dans un ordre arbitraire, et chacun indépendamment des autres peut avoir une valeur pour l'utilisateur.

Un récit utilisateur est une phrase simple dans le langage de tous les jours permettant de définir avec suffisamment de précision le contenu d'une fonctionnalité à développer. Elle contient généralement trois éléments descriptifs de la fonctionnalité, sous la forme :

En tant que <qui>, je veux <quoi> afin de <pourquoi>

Le *qui* indique l'utilisateur du point de vue duquel on se place. Il s'agit souvent

d'un rôle dans l'usage du produit . Pour mieux illustrer la diversité des besoins, on peut également utiliser le concept de [persona](#), c'est-à-dire une personne fictive, représentative de catégories d'usagers du produit, et à laquelle on peut s'identifier pour mieux comprendre ses attentes. L'identification et la description des personas se fait alors avant de commencer l'écriture des récits utilisateurs.

Le *quoi* décrit succinctement la fonctionnalité ou le comportement attendu. Le but du récit n'est pas d'en fournir une explication exhaustive. Cette dernière est obtenue de façon interactive au cours d'une conversation avec les utilisateurs concernés ou leurs représentants².

Le *pourquoi* permet d'identifier l'intérêt de la fonctionnalité et d'en justifier le développement. Il permet également de mieux évaluer la priorité du récit. Lorsque des points importants sont identifiés au moment d'écrire un récit, on peut faire une annotation. Ceci permet de garder la phrase simple, tout en gardant à l'esprit ces points².

Chaque récit utilisateur doit être complété avec des **critères d'acceptation**. C'est une liste d'éléments qui permettront à l'utilisateur de confirmer que la fonctionnalité, une fois livrée, correspond effectivement aux attentes². Le fait de définir ces critères à l'avance permet aussi d'assurer que la description de la fonctionnalité est suffisamment précise pour être réalisable.

Recueillir les exigences



- ✓ Analyse de documentations
- ✓ Brainstorming
- ✓ Entretiens
- ✓ Groupes de discussions, Ateliers
- ✓ Questionnaires, Enquêtes
- ✓ Analyse comparative, Benchmarking



Le recueil des exigences fait appel à un ensemble de techniques, qui font partie plutôt du domaine du consulting que de la Gestion de Projet :

Analyse de la documentation :

La documentation des systèmes existants, articles de presse, études comparatives, études de marché, forums Internet. Les

spécifications existantes contiennent souvent des exigences qui peuvent être répétées. Les spécifications fonctionnelles générales ou détaillées des logiciels commerciaux existants sont également une mine d'informations qui peuvent être réutilisées, après analyse, pour développer des exigences. Les fiches d'anomalies, rapports de tests, et rapports du helpdesk. Normes et documents existants chez des concurrents, ou d'autres entreprises dans un domaine comparable. Analyse de l'interface du système (recherche d'informations sur les interfaces avec les systèmes externes en interaction) Analyse des interfaces utilisateurs existantes.

Brainstorming :

Le brainstorming est une technique utilisée pour générer et collecter de multiples idées liées aux exigences du projet et du produit.

Entretiens :

Un entretien est une approche formelle ou informelle visant à obtenir des informations des parties prenantes en leur parlant directement. Elle est généralement réalisée en posant des questions préparées et spontanées

et en enregistrant les réponses. Les entretiens sont souvent menés sur une base individuelle entre un enquêteur et une personne interrogée. Les entretiens sont également utiles pour obtenir des informations confidentielles.

Groupes de discussion – Ateliers.

Les groupes de discussion réunissent des parties prenantes présélectionnées et des experts en la matière pour connaître leurs attentes et leurs attitudes à l'égard d'un produit, d'un service ou d'un résultat proposé. Un modérateur formé, guide le groupe dans une discussion interactive conçue pour être plus conversationnelle qu'un entretien individuel.

Questionnaires et enquêtes :

Les questionnaires et les enquêtes sont des séries de questions écrites conçues pour accumuler rapidement des informations auprès d'un grand nombre de répondants. Les questionnaires et/ou les enquêtes sont les plus appropriés avec des publics variés, lorsqu'un délai d'exécution rapide est nécessaire, lorsque les répondants sont

géographiquement dispersés et lorsqu'une analyse statistique peut être appropriée.

Analyse comparative :

Le benchmarking consiste à comparer les produits, processus et pratiques réels ou planifiés à ceux d'organisations comparables afin d'identifier les meilleures pratiques, de générer des idées d'amélioration et de fournir une base pour mesurer les performances. Les organisations comparées lors du benchmarking peuvent être internes ou externes.

Formaliser les exigences

- ✓ **Structurer**
- ✓ **Modéliser**
- ✓ **Prioritiser**



Ces exigences se formulent sous forme textuelle : le produit doit...

Exemple : le produit doit permettre de s'authentifier afin d'accéder à l'ensemble des commandes en cours

Dans l'approche prédictive, l'approche est libre et il s'agira de classer les exigences et les contraintes, le plus souvent dans un tableur,

ce qui permettra de les trier, de les filtrer et de les classer. Ces exigences formeront le **cahier des charges** qui pourra être le socle d'un **appel d'offres**. Ce document qui décrit dans le détail les attendus du projet est une base pour la planification, mais sera aussi utile pour clore le projet, quand il sera nécessaire de vérifier que le projet répond bien aux attentes (Phase de recette).

La formalisation de cas d'utilisation (use cases) est fréquente pour illustrer les attentes de chaque type d'utilisateurs.

Dans la méthode AGILE ou SCRUM, très utilisée dans le développement de logiciels et fonctionnant en mode itératif, les exigences seront exprimées sous la forme de User Storys :

Forme : **En tant que <qui>, je veux <quoi> afin de <pourquoi>**

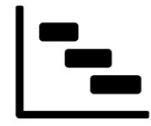
Le *qui* indique l'utilisateur du point de vue

duquel on se place. Il s'agit souvent d'un rôle dans l'usage du produit.

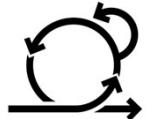
Le *quoi* décrit succinctement la fonctionnalité ou le comportement attendu.

Le *pourquoi* permet d'identifier l'intérêt de la fonctionnalité et d'en justifier le développement.

Les Users Stories sont stockées dans un Product Backlog (registre des Users Stories). Dans cette méthode, le classement et la priorisation des User Story sont décrit précisément. C'est grâce à ce classement que l'on sera en mesure de sélectionner quelles User Story feront partie des premiers cycles de réalisations. Ce Product Backlog fera l'objet d'un traitement itératif pour « Raffiner » au fil des cycles les Users Stories afin de les affecter à d'autres cycles de réalisation.



Exemple from MTI-PFEE: Exemple de TE



Agile Methodology: The Product Backlog

In an Agile approach, the product to be built is described by in a document Called "Product Backlog" that will live throughout the construction of the product.

The features we want to develop are described in the form of "user stories".

The detail of these user stories is refined according to their priority in terms of achievement and value for the project.

The most important and prioritarian are refined in precise and specific user stories , the needs that are less important can me formulated as a generic idea called EPIC.

The main difference with the classical approach is the fact that all the requirements are not fully specified and will be refined , restated iteratively during project execution .

Dans une approche Agile, le produit à construire est décrit par un document de « Product Backlog » qui va vivre tout au long de la construction du produit .

Les fonctionnalités que l'on souhaite développer sont décrites sous la forme de "user stories".

Le détail de ces user stories est raffiné en fonction de leur priorité en terme de réalisation et de valeur pour le projet.

Les plus importants et les plus prioritaires sont affinés dans des user stories précises et spécifiques, les besoins qui sont moins importants peuvent me formuler comme une idée générique appelée EPIC.

La principale différence avec l'approche classique est le fait que toutes les exigences ne

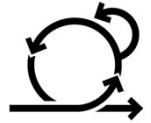
sont pas entièrement spécifiées et seront affinées, réformulées, précisées lors de l'exécution du projet.

C'est une approche itérative.

Différence AGILE:
On peut affiner au fur à mesure les exigences en avançant dans la Release

Attention ! Les US qui servent à planifier le premier SPRINT doivent être complètement détaillées.

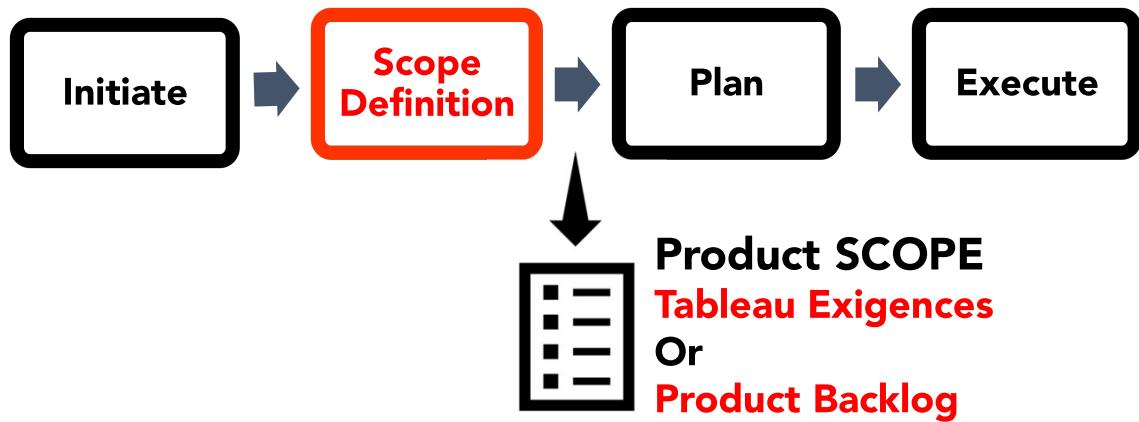
Niveau d'exigences équivalent à la méthode classique pour les premiers Sprints.



Exemples From MTI-PFEE

Exemple PB Commenté

Recall Presence Project context and Ask to students to express one functional requirement as a user story .



En résumé :

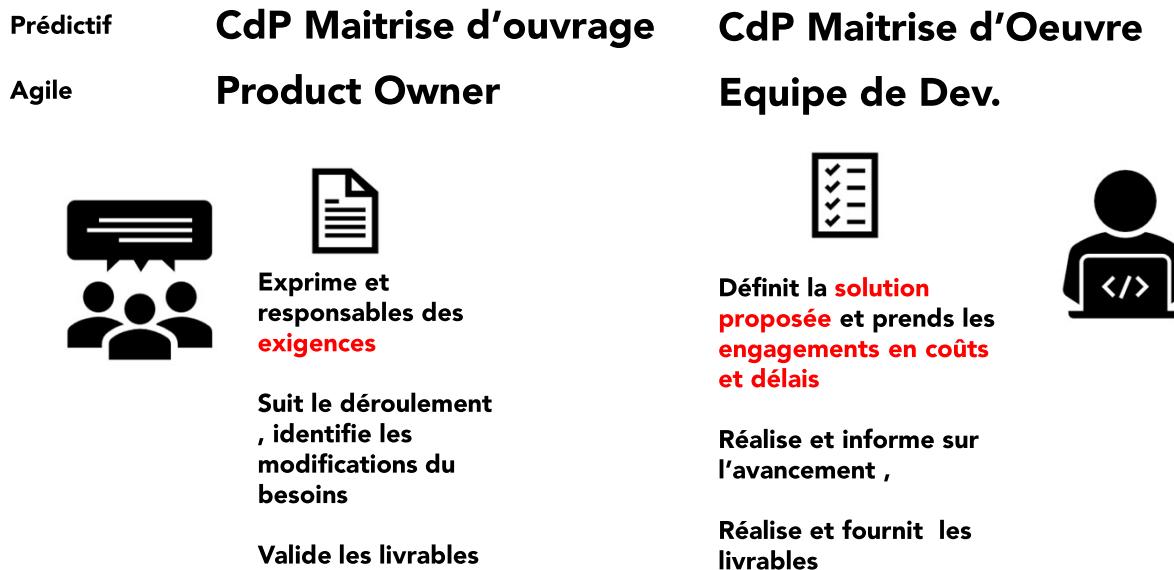
La Définition d'un Product scope (périmètre produit) est obligatoire avant de planifier le projet a fortiori avant de développer.

La documentation de ce scope doit être approuvée, partagée et suivie avec le client.

The product Scope is then essential . Normally the product scope is under Customer Team Responsabilty. in Agile methodology Product Scope is the Product Backlog managed by the Product Owner.

The project Scope will be defined in the planning phase of the life cycle and will express more the project deliverables we intend to provide

Les parties prenantes



Dans un modèle traditionnel, la relation est le contrat et l'attente du client est un engagement en termes de cout et de délai pour la réalisation. Ce modèle s'applique évidemment lorsque le demandeur « Maitrise d'ouvrage » fait appel à une société extérieure « Maitrise d'œuvre » pour la réalisation. Mais le plus souvent ce modèle client-fournisseur est également appliqué en interne d'une société lorsque la réalisation est confiée à un service interne (par exemple le service

informatique).

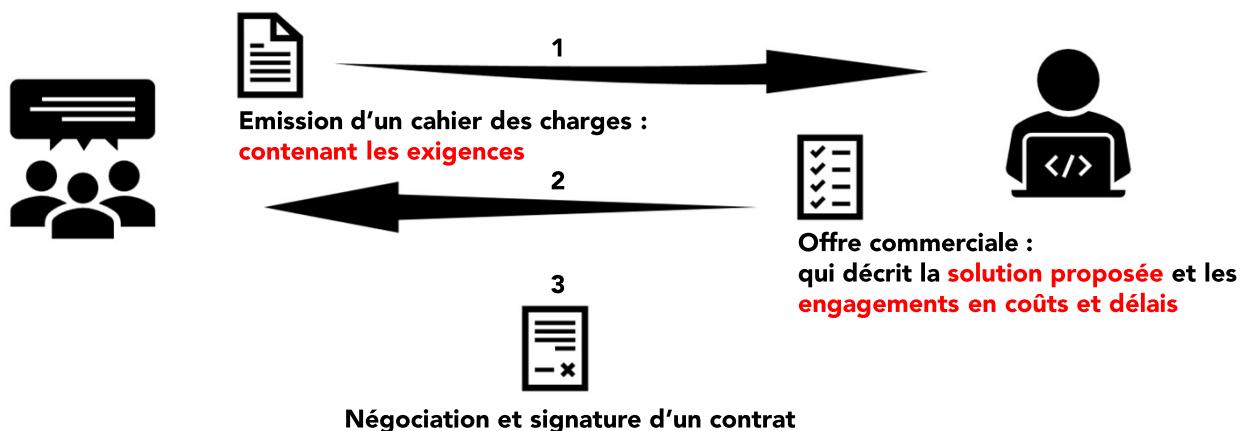
Dans ce modèle le client a dû établir un **cahier des charges** en renseignant de la façon la plus précise possible ses exigences et notamment ses **exigences fonctionnelles**.

En théorie dans ce modèle de relation, le cahier des charges doit être le plus complet et précis possible. En effet les changements, qui arriveront quand même, au cours du projet auront un impact en coût et en délai que le client souhaite éviter.

Ainsi, le plus souvent un long travail est réalisé pour décrire de façon détaillée tout le périmètre du projet. Avant la signature du contrat.

L'équipe de réalisation reçoit ce cahier des charges, et doit en faire l'analyse pour imaginer **la solution Produit** qui peut y répondre, et identifier **le travail à réaliser** puis l'organiser dans un « plan projet » qui constitue la réponse au cahier des charges.

Modèle Traditionnel :



Dans un modèle traditionnel, la relation est le contrat et l'attente du client est un engagement en terme de cout et de délai pour la réalisation. Ce modèle s'applique évidemment lorsque le demandeur « Maitrise d'ouvrage » fait appel à une société exterieure « Maitrise d'œuvre » pour la réalisation. Mais le plus souvent ce modèle client-fournisseur est également appliqué en interne d'une société lorsque la réalisation est confié à un service interne (par exemple le service informatique).

Dans ce modèle le client a du établir un **cahier des charges** en renseignant de la façon la plus précise possible ses exigences et notamment ses **exigences fonctionnelles**.

En théorie dans ce modèle de relation, le cahier des charges doit être le plus complet et précis possible. En effet les changements, qui arriveront quand même, au cours du projet auront un impact en coût et en délai que le client souhaite éviter.

Ainsi, le plus souvent un long travail est réalisé pour décrire de façon détaillée tout le périmètre du projet. Avant la signature du contrat.

L'équipe de réalisation reçoit ce cahier des charges, et doit en faire l'analyse pour imaginer **la solution Produit** qui peut y répondre, et identifier **le travail à réaliser** puis l'organiser dans un « plan projet » qui constitue la réponse au cahier des charges.

Allez sur wooclap.com et utilisez le code GMAQIR

Dans les propositions suivantes, laquelle décrit le mieux la différence entre un Product Backlog et un Cahier des Charges en méthode prédictif ?

1 Le document décrit les contraintes de l'application, en complément des exigences
2 Le document décrit les exigences prioritaires de façon précise  détaillée
3 Le document prévoit une hiérarchisation par priorité des exigences
4 Le document, à dessein, décrit de façon grossière les exigences ayant un faible niveau de priorité

Cliquez sur l'écran projeté pour lancer la question

wooclap

Allez sur wooclap.com et utilisez le code GMAQIR

Dans les propositions suivantes, laquelle décrit le mieux la différence entre un Product Backlog et un Cahier des Charges en méthode prédictif ?

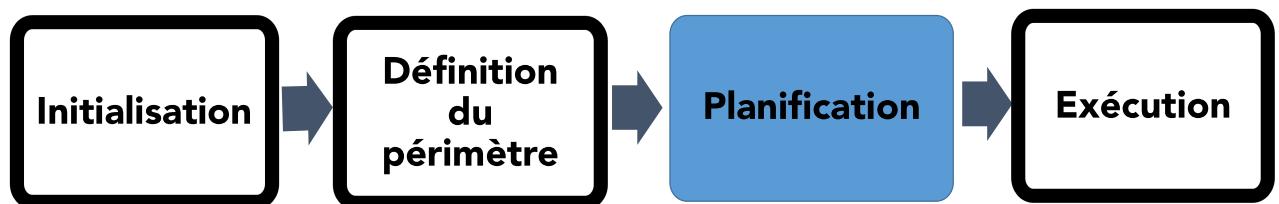
1 Le document décrit les contraintes de l'application, en complément des exigences
2 Le document décrit les exigences prioritaires de façon précise  détaillée
3 Le document prévoit une hiérarchisation par priorité des exigences
4 Le document, à dessein, décrit de façon grossière les exigences ayant un faible niveau de priorité

wooclap

ING02 – PM Refresh



Cycle de vie Projet:



La gestion des délais commence par la construction de l'échéancier dans la phase de **Planification** :



Du contenu projet à l'échéancier



La construction de l'échéancier projet (Schedule en anglais) fournit un plan détaillé qui indique comment et **quand, le projet fournira ses principaux livrables**, services ou résultats qui ont été définis dans le contenu du projet. Cet échéancier, par son caractère visuel, sera l'outil principal de communication des performances projets. Il servira à gérer les attentes des parties prenantes. Le plus souvent, il prendra la forme d'un diagramme de GANTT.

Dans **l'approche prédictive**, le processus qui permet de passer du contenu défini sous forme de WBS à l'échéancier est le suivant :

- * Le contenu est découpé sous la forme d'une hiérarchie de Lots de Travaux – C'est ce que nous avons vu dans la partie Contenu – qui contiennent l'ensemble du travail à réaliser dans le cadre de ce projet. Nous verrons que ces **lots de travaux** sont d'une bonne granularité pour effectuer le **suivi des coûts**. Mais ils ne sont pas assez détaillés pour procéder à l'estimation de charges.
- * Ces lots de travaux sont découpés en **activités** qui seront susceptibles d'être estimées en termes d'effort et de durée.
- * **Les dépendances** entre ces activités seront observées pour produire un enchainement logique.
- * Pour chaque activité, on pourra estimer **la charge de travail** nécessaire pour l'effectuer, et, selon les ressources qui

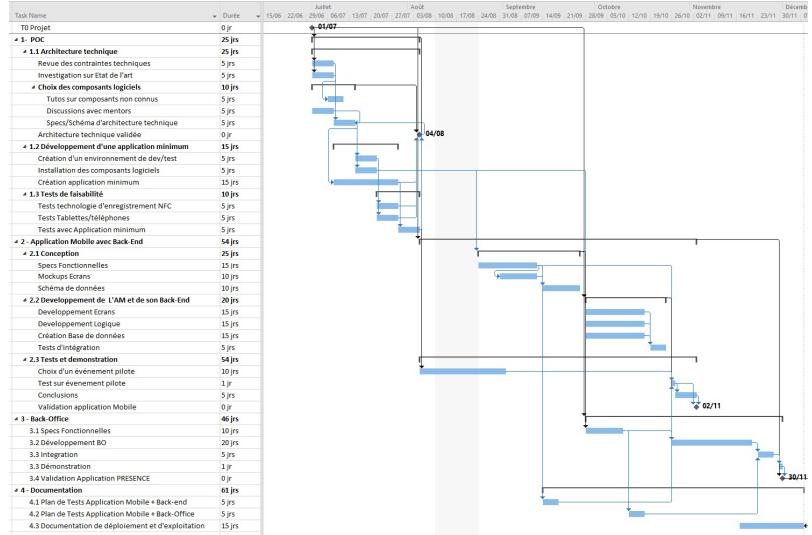
pourront y être affectées, **sa durée**.

- * Enfin un échéancier sera construit, qui propose un déroulement complet de ces activités, **des jalons** qui permettent de suivre le déroulement et définir les dates de livraison des principaux objets résultats du projet.

La planification prédictive ou classique, produit en général un échéancier sous forme de diagramme de GANTT.

Equivalences Anglaises: SCHEDULE
MANAGEMENT; SCHEDULE

Exemple de GANTT chart



WBS: Work Breakdown Structure Lots de Travaux (Work Packages)

TOUT (100%) le travail à faire dans le projet, y compris la gestion du projet, est décomposé, divisé et subdivisé en une STRUCTURE hiérarchique.

Cette structure est appelée **WBS : WORK BREAKDOWN STRUCTURE** (Arborescence de tâches).

Le terme anglais reste une référence incontournable en gestion de projet, même les ouvrages français font référence au terme WBS.

Le niveau le plus bas de la structure WBS est appelé **lot de travaux** (WORK PACKAGE).

Ils sont souvent associés à des **livrables** projets et représentent en général un niveau intéressant pour le suivi des coûts : **1 Lot de travaux = 1 ligne budgétaire.**

On notera qu'un lot de travaux aboutit généralement à un livrable qui est le fruit du travail réalisé. Cependant il faut voir que la notion de livrables peut se décomposer en

- * Livrables finaux qui sont la fourniture attendues par le client (le logiciel complet , un document d'exploitation...)
- * Livrables intermédiaires qui sont des étapes de réalisations qui permettent la bonne réalisation des étapes suivantes (par exemple une maquette, une version alpha du logiciel , ...)

C'est aussi à ce niveau que l'on peut choisir de confier un Lot de travail à un sous-traitant.

Suivant le métier ou le domaine professionnel, il existe des modèles de WBS qui peuvent être

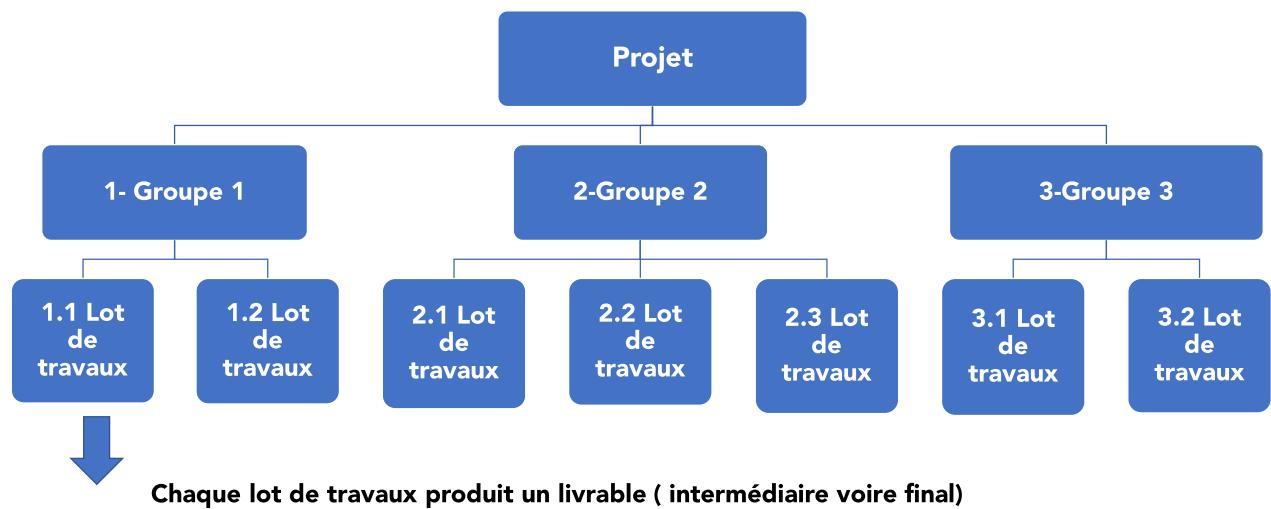
appliqués.

Une première approche consiste à :

- * Utiliser les phases du cycle de vie du projet comme premier niveau de décomposition : conception, réalisation, tests... Utiliser les principaux livrables comme deuxième niveau de décomposition : logiciel, documentation, support de formation etc...

Equivalences anglaises: WORKBREAKDOWN STRUCTURE; WORKPACKAGES; DELIVERABLES

Structure d'une WBS



La structuration de l'ensemble des travaux à réaliser s'organise en une arborescence souvent à 3 niveaux parfois plus.

Le principe est que le premier niveau est un regroupement logique qui est librement choisi par le Chef de projet.

Ce peut être une organisation suivant le cycle de production par exemple: Conception, Développement , Intégration/test, déploiement .

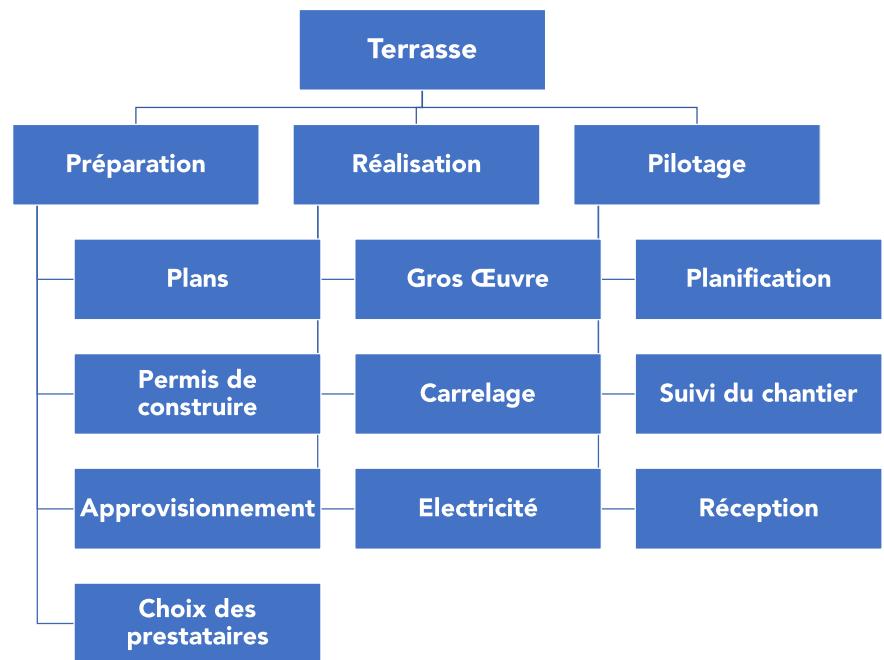
Mais d'autres organisations sont possibles, par exemple par fonction de l'entreprise : Etude, Marketing , Laboratoire, Industrialisation.

Le dernier niveau correspond à **des lots de travaux** : Chaque lot doit aboutir la production d'un livrable qui le plus souvent est intermédiaire : C'est-à-dire que la production est une étape dans la réalisation du produit final. Par exemple, la définition d'une architecture technique, la réalisation d'une documentation, l'approvisionnement de composants nécessaires à la réalisation...

Methode :

- **On recense l'ensemble des livrables à produire .**
- **On peut Scénariser la réalisation pour les identifier**

Exemple de WBS#2



Exemple de projet basique : construction d'une terrasse

La Décomposition des lots de travaux en activités :

- **Une taille qui permet l'estimation de charges**
- **S'appuyer sur l'expérience : Le jugement d'expert**
- **Pas de micro-tâches, impossible à manager**
- **Pour le développement logiciel : quelques jours**

Le découpage en activités consiste à décomposer les Lots de travaux de la WBS en activités qui permettent de produire ces lots de travaux. De la même façon que pour la WBS, il s'agit d'une décomposition du travail en éléments plus petits. La taille de ces activités élémentaires dépend du type de projet et est fortement influencée par la connaissance du travail à effectuer. Il faudra choisir la bonne taille d'activité pour permettre l'estimation de charges la plus précise possible : quelques jours en général,

et de pas tomber dans des micro-taches qui deviendraient, par leur nombre, impossible à manager et à planifier.

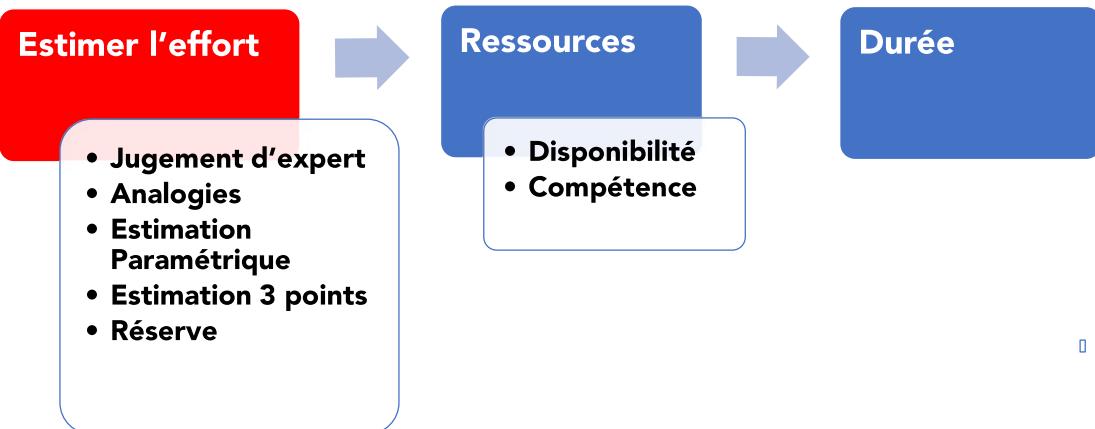
Vous pouvez examiner tous les enseignements tirés de projets similaires réalisés par votre entreprise pour avoir une bonne idée de ce que vous devez faire pour le projet actuel.

Un jugement expert, sous la forme de la contribution de membres de l'équipe de projet ayant une expérience préalable de l'élaboration d'énoncés de la portée du projet et de la WBS, peut vous aider à définir les activités.

Si l'on vous demande de gérer un projet dans un nouveau domaine technologique, vous pouvez également faire appel à des experts dans ce domaine particulier pour vous aider à définir les tâches afin de comprendre quelles activités seront impliquées. Vous pouvez créer une liste d'activités, puis demander à l'expert de l'examiner et de suggérer des modifications. Vous pouvez également impliquer l'expert dès le début et demander à

avoir une conversation sur la définition des activités avec lui avant même de faire votre première ébauche de la liste.

Charges/Ressources/Durée



L'estimation des charges est l'activité la plus importante de la construction de l'échéancier :

- * Elle produira des estimations sous forme de plages de durée,
- * Elle s'appuie sur des hypothèses qui devront être documentées,
- * Elle fait appel à un certain nombre de méthodes, décrites ci-dessous

Le jugement expert : il viendra des membres de votre équipe de projet qui connaissent bien le travail à effectuer. Si vous n'obtenez pas leur avis, le risque est grand que vos estimations soient erronées.

La méthode dite **DELPHI** consiste à organiser la consultation des experts de façon à clarifier la demande mais surtout à essayer d'obtenir un consensus par itération successives.

Par exemple on soumet à 4 experts une estimation de l'effort pour réaliser un logiciel, après leur première réponse on leur indique la moyenne obtenue et on leur demande d'évaluer à nouveau l'effort. On peut espérer alors converger, sinon on peut utiliser l'estimation en trois points sur les résultats obtenus.

L'estimation par analogie consiste à examiner les activités de projets antérieurs similaires à celui-ci et à déterminer le temps qu'il a fallu pour effectuer un travail similaire auparavant.

L'estimation paramétrique consiste à introduire des données sur votre projet dans une formule, une feuille de calcul, une base de données ou un programme informatique qui produit une estimation. Le logiciel ou la formule que vous utilisez pour l'estimation paramétrique est construit sur une base de données de durées réelles de projets antérieurs. Par exemple, on utilisait dans les années 80 des abaques qui donnaient combien de ligne de code pouvaient être développées par jour. On peut avoir un nombre d'heures par écran par exemple. Un autre système de même nature est / était employé par des entreprises pour le développement logiciel, **les points de fonction**, le principe général consistait à évaluer uniquement l'ensemble des fonctionnalités attendues par l'utilisateur au travers d'une notation. La mesure de cette richesse fonctionnelle pouvait ensuite être traduite suivant les pratiques et l'expérience de l'entreprise en quantité d'effort de développement. On notera que cette idée est très similaire à l'attribution des points dans

l'estimation d'un sprint dans la méthode AGILE.

L'estimation en trois points consiste à obtenir trois chiffres : une estimation réaliste qui a le plus de chances de se produire, une estimation optimiste qui représente le meilleur scénario et une estimation pessimiste qui représente le pire scénario. L'estimation finale est la moyenne.

Plus probable (tM)

Optimiste (tO)

Pessimiste (tP)

$$tE = (tO+tM+tP)/3$$

L'analyse des réserves consiste à ajouter du temps supplémentaire au calendrier (appelé réserve pour imprévus ou tampon) pour tenir compte des risques supplémentaires.

Les Dépendances



Une fois cette liste d'activités établies, il convient de modéliser **dans quel ordre** elle doivent être effectuées. Pour cela on ajoutera à leur attributs des prédecesseurs et des successeurs. On appliquera la méthode des antécédents ou PDM (Precedence Diagramming Method)

Une fois cette liste d'activités établies, il convient de modéliser **dans quel ordre** elles doivent être effectuées. Pour cela on ajoutera à leur attribut des prédecesseurs et des successeurs. On appliquera la méthode des antécédents ou PDM (Precedence Diagramming Method)

Les principales liaisons entre les activités sont de type Fin-Début (Finish to Start), mais on observera aussi des relations Finish to Finish ou Start To Start.

- * **Liaison Fin-Début** : le plus commun.
L'activité B ne peut pas commencer tant que A n'est pas terminé. Par exemple, l'installation de l'OS sur un PC ne peut pas débuter tant que le PC n'est pas monté.
- * **Liaison Fin-Fin** : B ne peut pas se terminer avant la fin de A. par exemple la documentation ne peut pas être terminée tant que le code n'a pas été complètement développé.
- * **Liaison Début-Début** : B ne peut pas commencer tant que A n'a pas au moins commencé. Les réunions de suivi projet ne peuvent pas débuter avant que la phase de conception détaillée n'ait débuté.

Ces relations s'accompagnent souvent d'un délai **d'avance (lead)** ou **de retard (lag)**.

Par exemple :

La livraison suivra la commande avec un délai de 6 semaines : **Lag**

Le développement suivra l'approbation des spécifications mais peut être anticipé d'une semaine : **Lead**

L'ensemble de ces relations permettra d'établir un **diagramme de précédence**. Cette forme de diagramme est peu utilisée aujourd'hui, et ces relations de précédences seront en général visualisées dans le diagramme de GANTT.

Allez sur wooclap.com et utilisez le code **GMAQIR** 

Je construit ma terrasse : le maçon a besoin de matériaux que je commande chez Point P.
La tache A "Approvisionnement" dure 1 semaine (5J). la tache B "Réalisation gros œuvre Terrasse" dure..



- ① B est en Fin Début par rapport à la tache A (fin de A déclenche B).  0% | 0 people
- ② B est en Début Fin de la tache A (début de B déclenche A).  0% | 0 people
- ③ B est en Début Début de la tache A avec un retard de 5 jours (A démarre et B démarre 5 jours plus tard).  0% | 0 people
- ④ B est en Fin Fin de A avec 2 semaines d'anticipation. la fin de B conditionne la Fin de A mais 5 jours avant.  0% | 0 people

Cliquez sur l'écran projeté pour lancer la question

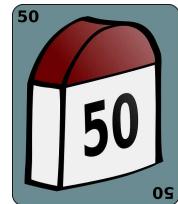
wooclap

lock search 105% refresh

0 / 0 people

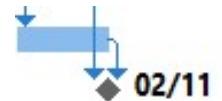


Les Jalons (Milestones)



Représentent les évènements importants d'un projet:

- Jalons de livraison des principaux livrables
- Jalons de validation (spécifications)
- Jalons d'acceptation
- Jalons de synchronisation avec des évènement externes
- Jalons de Facturation



Les Jalons, représentés par des losanges sur le Diagramme de GANTT, sont, comme les bornes sur une route, des points de repères pour contrôler l'avancement du projet et pour synchroniser les principales activités. Ils représentent des évènements importants d'un projet

Représentés par une activité de durée 0, ils permettent de souligner l'importance d'un événement dans le Projet.

On construira une **liste des jalons** que l'on classera par types. Parmi les plus courants :

- * Jalons de livraison
- * Jalons de validation
- * Jalons de recette ou d'acceptation
- * Jalons de facturation
- * Jalons de Synchronisation avec des projets ou éléments externes

La construction de l'échéancier

- Logiciel de gestion de projet : MS Project ou GanttProject
- Saisie des activités avec leurs attributs: Précédence, durée, ressources
- Saisie des Jalons
- Construction du Diagramme de Gantt
- Il s'agit de l'échéancier initial : BASELINE

A partir de la liste des **activités**, qui ont maintenant une **estimation de durée et des informations de prédecesseurs/successeurs**, on peut construire un échéancier en utilisant un logiciel de Management de projet comme MS Project ou GANTT Project (gratuit).

Ces logiciels utilisent les données des activités et des jalons fournies en entrée pour construire un diagramme de Gantt.

La construction de l'échéancier est en général un processus itératif que l'on devra revoir avec l'équipe projet pour s'assurer que les activités confiées à chaque ressource ont bien été synchronisées.

La méthode du **chemin critique** est la méthode utilisée pour estimer la durée minimum du projet et déterminer le degré de flexibilité de l'échéancier sur les chemins de réseau logique. Le chemin critique est la séquence d'activité qui représente le chemin le plus long du projet et qui déterminent la durée la plus courte possible de ce projet. Ce chemin critique est visualisé automatiquement par les logiciels comme Microsoft Project.

Pour agir sur la durée totale du projet, on pourra essayer d'agir sur la séquence d'activités qui sont sur le chemin critique, en ajoutant des ressources par exemples. On appelle cela une technique de compression des délais. On peut aussi choisir de paralléliser des tâches qui devraient logiquement avoir lieu en séquence.

On saisira dans ce logiciels la liste des **jalons**.

Top Down vs Bottom Up

L'approche proposée : décomposition en activités et estimation de chaque activité est appelée estimation BOTTOM-UP (du bas vers le haut).

La méthode est précise, mais génère souvent une **SURESTIMATION** (tendance à surestimer légèrement la durée et l'effort de chaque activité).

Pour contrebalancer l'estimation ascendante on utilise une approche descendante ou TOP-

DOWN : Sur la base de l'expérience, il s'agit d'obtenir une estimation de la durée ou de l'effort d'une phase complète du projet ou d'un lot de travail complet dans un projet passé qui a réalisé des travaux de même nature, et d'analyser comment le programme de travail et les activités peuvent correspondre à cette estimation.

Souvent, cette nouvelle itération de la construction de l'échéancier permettra de se rapprocher d'une meilleure planification de base.

La Timeline ou Chronologie

- Plus synthétique que le GANTT
- Présente les principales phases
- Expose les jalons principaux
- Utilisée pour une présentation en comité de pilotage

Pour communiquer l'avancement de son planning avec les parties prenantes, le diagramme de GANTT est souvent trop détaillé pour être partagé dans une présentation.

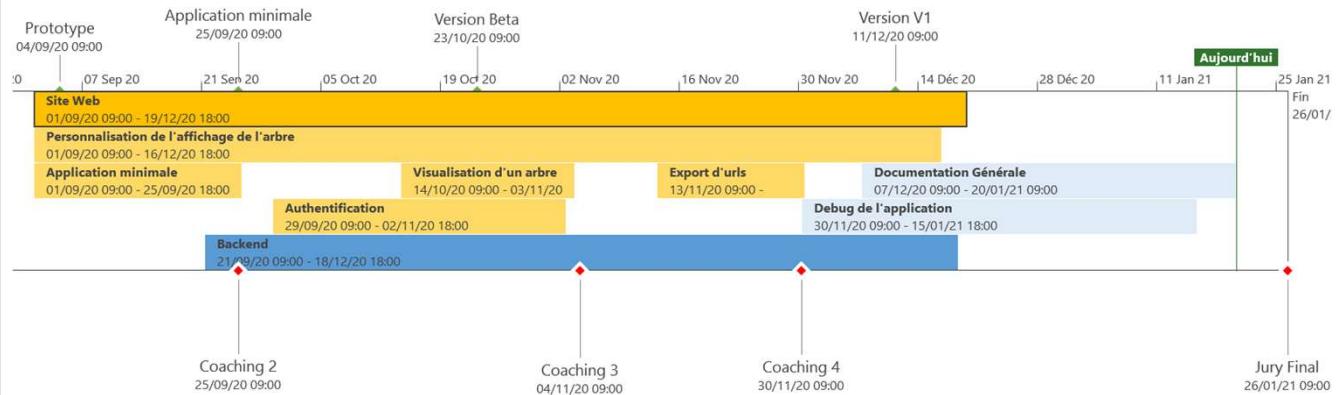
On choisit alors une représentation plus synthétique sous forme de Timeline.

Le logiciel Microsoft Project propose la construction de timeline synthétique à partir des données du projet enregistré, c'est

probablement la meilleure solution mais si on ne dispose pas de Microsoft Project on peut choisir une timeline parmi les modèles existants sur Internet ou disponibles dans Power point.

Exemples from MTI PFE Timelines

Timeline





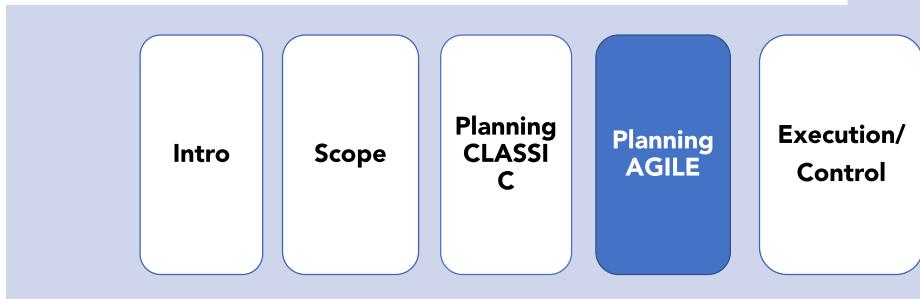
En résumé :

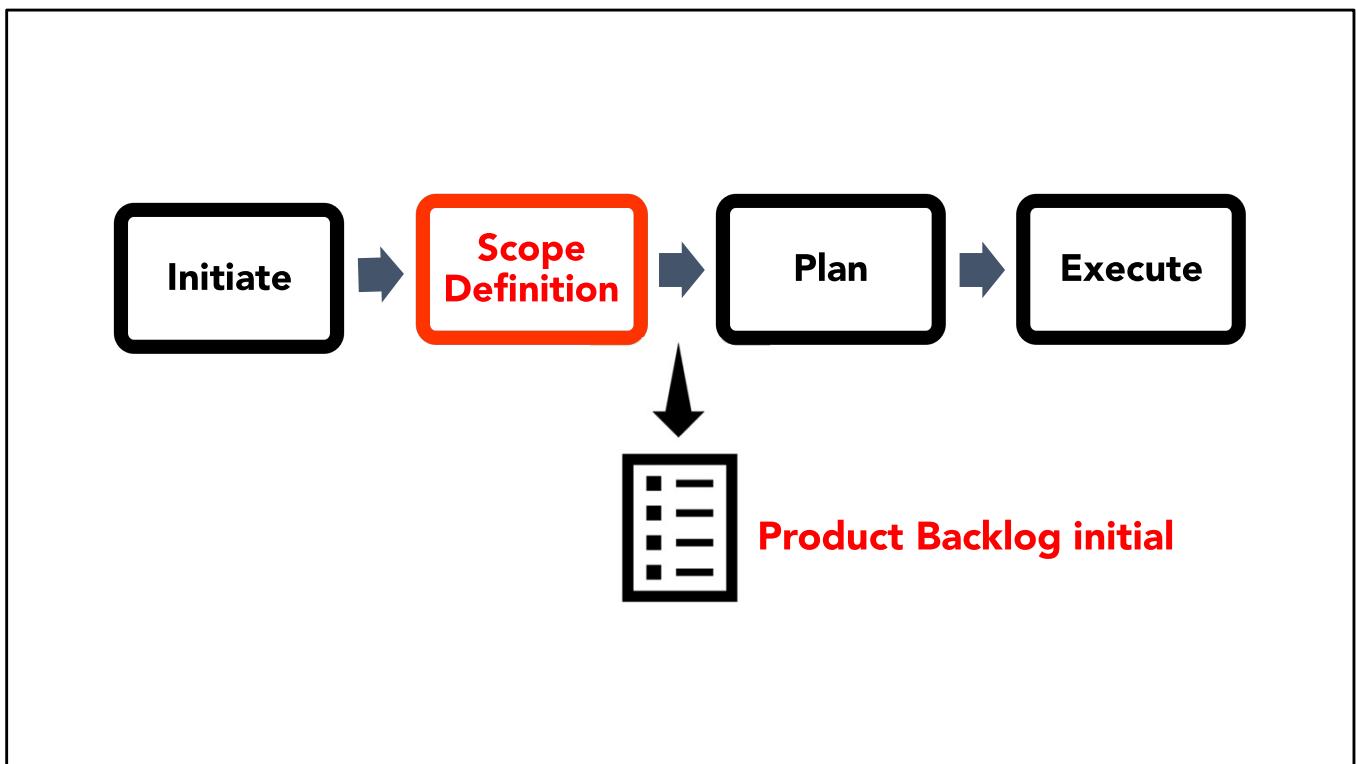
En découplant l'ensemble des productions à réaliser et le travail qu'elles représentent,

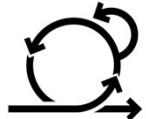
puis, après estimation, en répartissant dans le temps ces activités,

vous disposez d'un plan et d'un planning initial (BASELINE) de réalisation de votre projet complet.

ING02 – PM Refresh





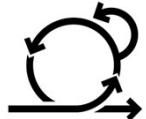


Set Priorities

Affecter des priorités

Thanks to the order settings of the Product back log, we have grouped the user stories which are have high dependancies , we have determined the ones that bring the highest value. Then we have just to give a priority ranking based on this information. We will have then on the top of the product backlog the stories or group of stories that have the highest priority up to the bottom of the P.B with low priority stories. Most of the time the highest priority user stories are precisely stated and give a detailed description of what is expected. On the other end , at the bottom of P.B. we will just find EPICS or even ideas of new features .. Product Owner is the only person that update the P.B.

Grâce à l'ordonnancement du P.B., nous avons regroupé les user stories qui sont ont des dépendances élevées, nous avons déterminé celles qui apportent le plus de valeur. Alors nous avons juste à procéder à un classement en affectant des priorités basées sur ces informations. Nous aurons alors en tête du P.B. les stories ou le groupe de Stories qui ont la plus haute priorité et en bas du P.B les stories à faible priorité. La plupart du temps, les U.S. les plus prioritaires sont énoncées avec précision et donnent une description détaillée de ce qui est attendu. Par contre , au bas du P.B. nous aurons juste des EPICS ou même des idées de nouvelles fonctionnalités .. Le Propriétaire est la seule personne qui met à jour le P.B.



Set an order to User stories Ordonner les User Stories

The first task is to order the Product Backlog. The principle is to detect the part of the product that brings the most value and order them based on this criteria

This P.B. scheduling can use different methods

- Simple methods such as MOSCOW :
 - Mo — Must have: must be achieved.
 - S — Should have: should be carried out if possible.
 - Co — Could have: could be achieved if there is no impact on other tasks in progress.
 - W — Won't have: will not be realized right away but would be desirable
- Most complex: For instance “Story mapping” that combines the analysis of the relationships between stories, the analysis of the impact on the business, This may lead to a product roadmap

Priorities will be deduced and these priorities will contribute to the ordering of the Product Backlog, but other features (e.g. the expected availability date) may also be included in the ordering of the Product Backlog.

Ordering the PB is part of the PB refinement. We put in place an initial order that will be reviewed regularly, especially in the PB Grooming process.

La première tache consiste à ordonner le Product Backlog. L'idée générale consiste à

essayer de déterminer ce qui a le plus de valeur pour dans le produit pour les utilisateurs et d'ordonner suivant ces critères .

Cet ordonnancement du P.B. peut utiliser différentes techniques

- *les plus simples type MOSCOW :*

Mo — Must have : doit être réalisée.

S — Should have : devrait être réalisée si possible.

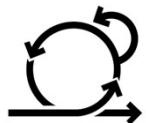
Co — Could have : pourrait être réalisée s'il n'y a pas d'impact sur les autres tâches en cours.

W — Won't have : ne sera pas réalisée tout de suite mais serait souhaitable

- *Des plus complexes : Story mapping qui mêlent l'analyse des relations entre les story , l'analyse de l'impact sur le business pour aboutir à une roadmap du produit*

On va en déduire des priorités et ces priorités vont contribuer à la mise en ordre du Product Backlog, mais d'autres caractéristiques (par exemple la date de disponibilité attendue) peuvent aussi rentrer en compte dans la mise en ordre du Product Backlog.

Le fait d'ordonner le PB participe au PB refinement. Nous mettons en place un ordre initial qui sera revu régulièrement notamment dans le process de PB Grooming.



Release Carving Découpage en Releases

Quelles sont les fonctionnalités essentielles que l'on doit mettre en Release 1?

A method often used for planning is to plan first the release : What are the scope of the user stories that shall be in the release ? And then split the release into sprints.

Release : période de temps constituée d'une série de sprints aboutissant généralement à déployer une version du produit.

Une méthode utilisée fréquemment consiste à planifier ce qui est dans la release : Quelles sont les US qui y sont nécessaires . Ensuite de couper en sprint la release .



Assign « Story Points » Affecter des « Story points »

1,2,3,5,8,13,21

For each User Story, the development team will then assign a number of points: This is an estimate of the difficulty in implementing this story. The difficulty includes **complexity, risk and effort**. It is above all a **relative** estimate between the different user stories.

Scales are used to differentiate between these difficulty levels:

Example: Fibonacci Suite: 1.2,3,5,8,13,21

The idea is that it is not the Product owner, nor the Scrum Master who defines these points but directly the development team with the search for consensus among the members.

This initial assignment will be reviewed very regularly during project execution

Ce processus vise à qualifier dans le product backlog la difficulté de développer chacune des user story.

Pour chaque User Story, L'équipe de développement va lui affecter un nombre de points: C'est une estimation de la difficulté à implémenter cette story. La difficulté inclus la complexité, le risque et l'effort.

C'est surtout une estimation relative entre les différentes user stories.

This process aims to qualify in the product backlog the difficulty of developing each user story.

On utilise des échelles qui permettent de bien différencier ces niveaux de difficulté:

Exemple: Suite de Fibonacci: 1,2,3,5,8,13,21

L'idée est que ce n'est pas les Product owner , ni le Scrum Master qui définisse ces points mais directement l'équipe de développement avec la recherche d'un consensus entre les membres

Cette affectation initiale sera revuee très régulièrement lors de l'exécution du projet.

Allez sur wooclap.com et utilisez le code **GMAQIR** 

La User Story 15 requérait 2 nouveaux écrans et la création de 2 End Points : elle a été estimée à 8 points.
la US 14 consistait à afficher un écran informatif : estimation 1 point 



Votez !

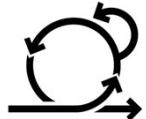
réponses reçues

wooclap

lock search 105 % refresh

0 users





Sprint = Timebox < 1 mois

Définition contenue dans scrumguide.org :

The Sprint: The heart of Scrum is a Sprint, a **time-box of one month or less** during which a "Done", useable, and potentially releasable product Increment is created. Sprints have consistent durations throughout a development effort. A new Sprint starts immediately after the conclusion of the previous Sprint.

During the Sprint:

- No changes are made that would endanger the Sprint Goal;
- Quality goals do not decrease; and,
- Scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned.

Each Sprint may be considered a project with no more than a one-month horizon. Like projects, Sprints are used to accomplish something. Each Sprint has a goal of what is to be built, a design and flexible plan that will guide building it, the work, and the resultant product increment.

Sprints are limited to one calendar month. When a Sprint's horizon is too long the definition of what is being built may change, complexity may rise, and risk may increase. Sprints enable predictability by ensuring inspection and adaptation of progress toward a Sprint Goal at least every calendar month. Sprints also limit risk to one calendar month of cost.

Le cœur de Scrum est le Sprint, qui a une boîte de temps (time-box), une durée, d'un mois ou moins au cours de laquelle un Incrément Produit « Fini » fonctionnel et potentiellement publiable est créé. Les sprints ont une durée cohérente durant la phase de développement. Un nouveau Sprint commence immédiatement après la conclusion du Sprint précédent.

Pendant le Sprint:

- *L'objectif du sprint est fixe ; les changements qui le remettent en cause ne sont donc pas permis ;*
- *Les objectifs de qualité sont maintenus ; ils ne sont jamais revus à la baisse ; et,*
- *Le périmètre peut être clarifié et renégocié entre le Product Owner et l'équipe de développement selon ce que l'équipe Scrum apprend.*

Chaque sprint peut être considéré comme un projet n'ayant qu'un horizon d'un mois. À l'instar d'un projet, un Sprint est utilisé pour accomplir quelque chose.

Chaque Sprint a un objectif de ce qui doit être construit, une conception (design) et un plan flexible qui guidera la construction, le travail lui-même et l'incrément produit résultant.



Découpage en Sprints

Meilleure combinaison de :

- **Priorité / valeur (à faire en premier)**
- **Capacité de l'équipe (vitesse)**
- **La logique technique**
- **Cohérence du PI**

- First step: define the duration of a sprint - according to the capacity of the development team.

Example: 1 month - 4-month project so 4 Sprints.

Be careful, each Sprint must produce an increment (PI) and therefore the team must have time to design, realize, and TEST within a Sprint time and the result shall be validated by the customer.

So do not make sprint very short Especially for students who do not work full time on the project ...

- Step Two: Define the content of each sprint Each sprint must contain about the same number of points.
 - A Sprint must produce an Increment (PI), which is a working product.
 - Sprints must also take into account the product backlog and the dependence between the U.S.

Première étape: définir la durée d'un sprint – en fonction des capacités de l'équipe de développement.

Exemple: 1 mois – Projet de 4 mois donc 4 Sprints.

Attention , il faut que Chaque Sprint produise un increment (PI) et donc il faut que l'équipe ait le temps de concevoir, réaliser , et TESTER au sein d'un Sprint et présenter/ faire valider le resultat par le client . Donc pas de sprint très court non plus surtout pour des Etudiants qui ne travaillent pas a temps plein sur le projet...

Deuxième étape: définir le contenu de chaque sprint

Chaque sprint doit contenir à peu près le même nombre de points.

Un Sprint doit produire un Increment (PI), qui est un produit fonctionnant.

Les Sprint doivent aussi prendre en compte l'ordonnancement du Product Backlog et les dépendance entre les U.S.

En résumé :

Vous avez convenu dans le Product-Backlog le périmètre de la release à réaliser.

Vous avez décomposé cette release en N Sprints d'un poids équivalent pouvant être réalisés successivement, a priori, dans le temps fixé pour un sprint.

Exemple de Product Backlog ordonné

Des prérequis avant de commencer le développement ?

Le Sprint 0.

Sprint 0: Get some technical & quality items on the Product Backlog, Provide a minimal environment that enables the writing of quality code, and. Write a piece of real code, no matter how small.

This shall be a quite small time unit where the dev team defines the technical architecture of the application and sets up this development environment .

Il s'agit de définir un socle minimum avec l'architecture produits, les composants logiciels, l'environnement de développement qui va permettre de dérouler des sprints.

Notion de ENABLER (SAFe)

Des prérequis avant de commencer le développement

:

Organisation de l'équipe de développement :

Team charter



Evidemment mais encore peut-on le préciser pour qu'une équipe fonctionne bien, il faut que quelques règles soient adoptées par tous. Pour des équipes importantes, on peut établir formellement un Team Charter qui fixe ces règles. Quoiqu'il en soit, il conviendra de discuter ensemble de quelques sujets pour organiser le travail et les thèmes peuvent être :

- Roles and responsibilities :

Le Product Owner

Le Product Owner est responsable de maximiser la valeur du produit résultant du travail de l'équipe de développement. Le Product Owner est le seul responsable de la gestion du Product Backlog.

Le Scrum Master

Le Scrum Master est chargé de promouvoir et supporter Scrum. Les Scrum Masters remplissent leur rôle en aidant tout le monde à comprendre la théorie, les pratiques, les règles et les valeurs de Scrum. Le Scrum Master est un leader-serviteur de l'équipe Scrum. Le Scrum Master assiste les personnes externes à l'équipe Scrum pour identifier quelles sont les interactions bénéfiques avec elle. Le Scrum Master aide tout le monde à adapter leurs interactions avec l'équipe Scrum pour maximiser la valeur créée par cette équipe.

L'équipe de Développement

L'équipe de développement se compose de professionnels qui fournissent un incrément « Fini » potentiellement publiable (Releasable) à la fin de chaque Sprint.

- Values & behaviours: assez américain mais pourquoi pas en discuter
- meeting details: when , how , documentation ...
- technical practices:
 - repository (GITHUB, GITLAB...).... ,
 - Tests tools
 - ...
- communication tool:
 - Interpersonal communication tools (slack , skype..)
 - Activity tracking (Trello, GITHUB Project , JIRA...) ;

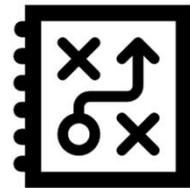
Rappel :

**Un sprint vise à réaliser un Product Increment (PI) :
qui est un élément de code fonctionnel.**

**Un sprint n'est pas une phase de définition de
besoin, ou une étape de mise en place de moyens
techniques**

**Le PB est maintenant réparti en
Releases et découpé en Sprints.
Démarrons avec le Sprint 1!**

Chaque SPRINT commence avec : **Le SPRINT PLANNING:**



Etablir le plan de travail de chaque membre de l'équipe pour la durée du Sprint !

The start of a Sprint will be achieved by a Sprint Planning meeting that aims to define the work to be done by the team to assign it to plan it over time. This meeting includes the entire team and optionally the Product Owner in order to clarify expectations. The Sprint Planning meeting is quite long but max 8hours for a one-month sprint.

How it works:

- Explanation of stories that are in the sprint to the team
- Defining the conditions for a job to be considered finished?? So how is the realization tested?
- Review and confirmation of points awarded to the user story by Dev's team.
- For each user story: We cut into task to carry out the User Story
 - Cutting the work that will be assigned to each developer of the team:
 - Possibly estimating the duration of the load assignment to team members (predict tests, bug fixes)
 - Ordering work (what you do first, and then as well as managing dependancies)
- Review of the consistency between the Us weighting points and the dev team's sprint work plan: Possible adjustment : adding or removing user story from Sprint Backlog
- Putting in a Scrum Board TO DO list of the US decomposed in task: Scrum Board can be supported by collaboration tool GITHUB PROJET, Trello ...

At the end of Sprint Planning, we know the features we will achieve, how to test them,

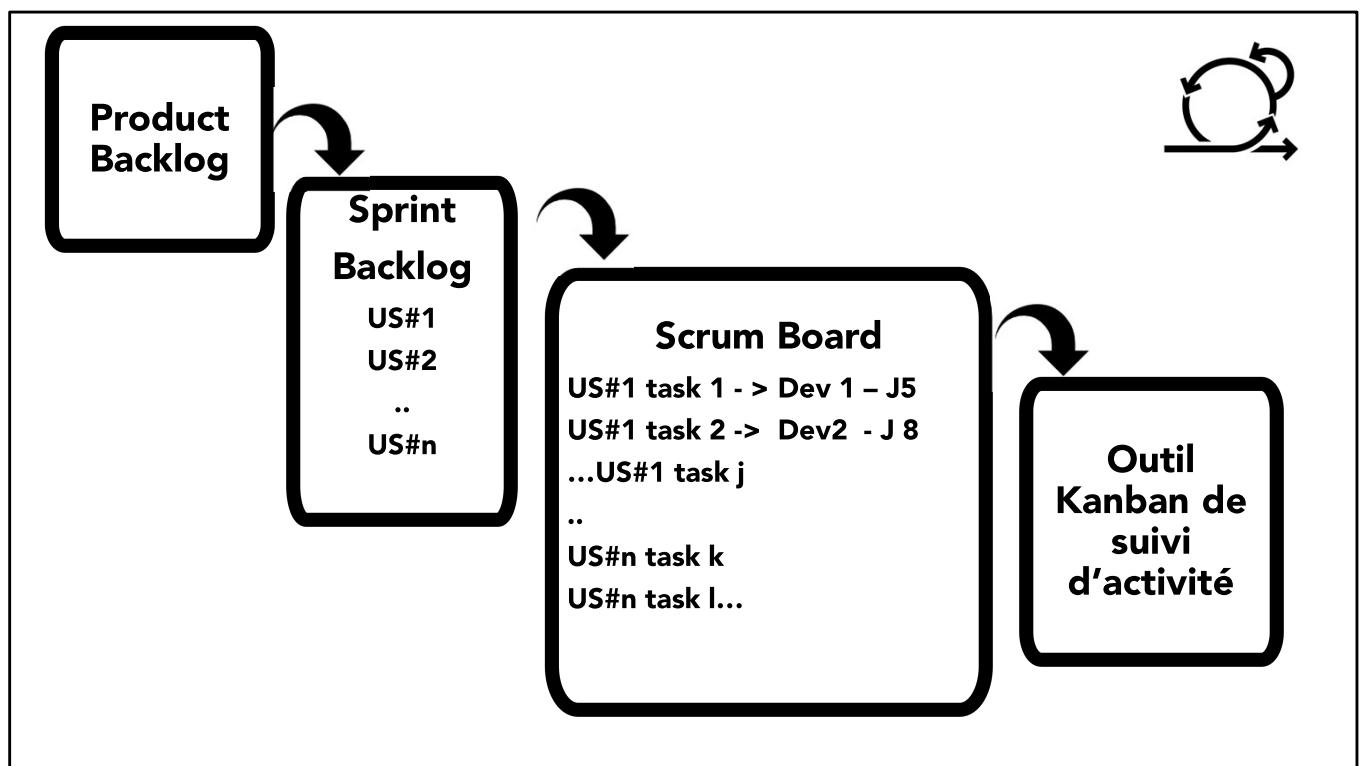
how we organize the work of the dev team to achieve the result.

Le démarrage d'un Sprint va être réalisé par une réunion de Sprint Planning qui vise à définir le travail à réaliser par l'équipe à l'affecter à le planifier dans le temps. Cette réunion comprend l'ensemble de l'équipe et optionnellement le Product Owner pour clarifier les attentes .

La réunion de Sprint planning est assez longue mais max 8h pour un sprint d'un mois.

Le déroulement :

- *Explication a l'équipe de dev des stories qui sont dans le sprint*
- *Définition des conditions pour qu'un travail soit considéré comme terminé /fini ?
Donc Comment est testée la réalisation ?*
- *Revue et confirmation des points attribués à la user Story par l'équipe de Dev.*
- *Pour chaque user story : On découpe en activité à réaliser la user story Découpage des travaux qui vont être affectés à chaque développeur de l'équipe :*
 - *Possiblement estimation de la durée de réalisation des taches*
 - *affectation aux membres de l'équipe (prévoir les tests, les corrections de bug)*
 - *Ordonnancement des travaux (ce que l'on fait en premier, puis ensuite et gestion des dépendances)*
- *Revue de la cohérence entre la pondération des US et le plan de travail de l'équipe de dev sur le sprint : Ajustement éventuel ajout ou suppression de user story du Sprint Backlog*
- *Mise dans un Scrum Board colonne TO DO des US decomposée en tâche: Scrum peut être supporté par outil de collaboration GITHUB PROJET, Trello ...*
- *En fin de Sprint Planning, on sait les fonctionnalités que l'on va réaliser, comment les tester, comment on organise le travail de l'équipe de dev pour arriver au résultat.*

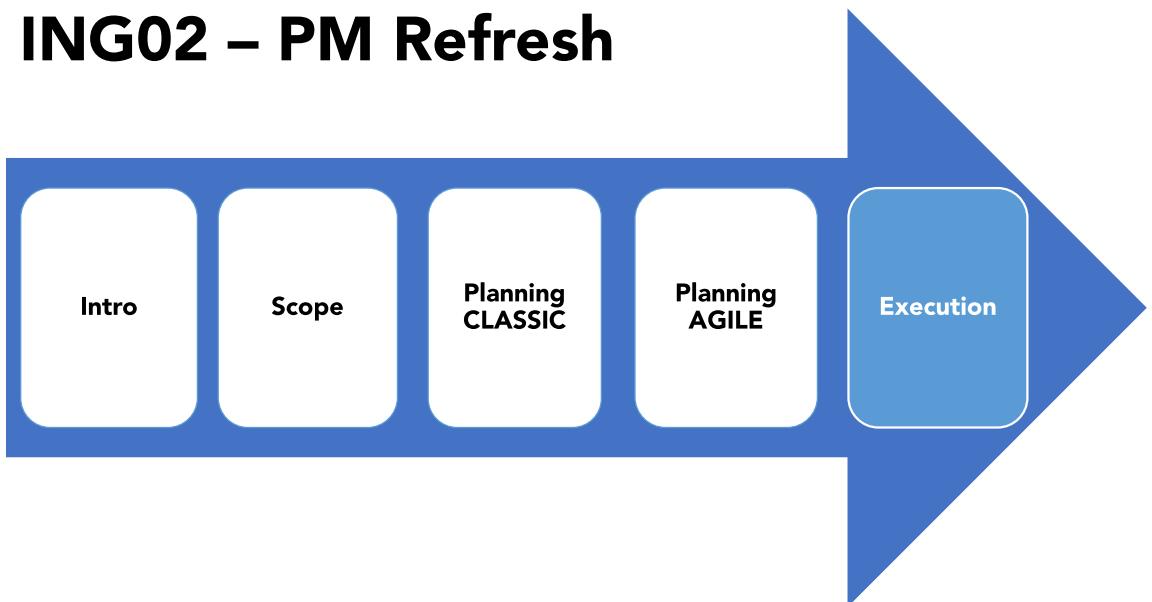


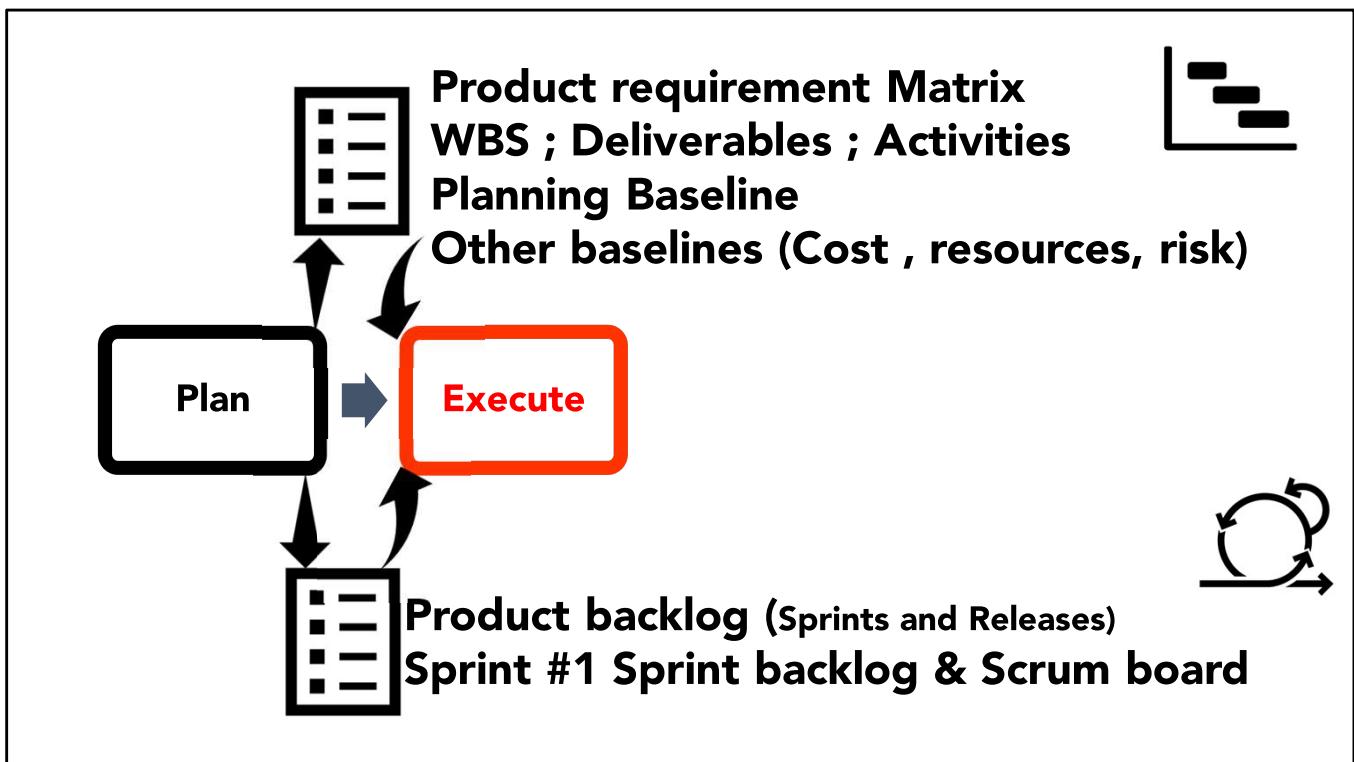
Exemple de Trello

En résumé :

Vous avez un plan et un planning détaillé pour toutes les activités à mener lors du sprint que vous entamez.

ING02 – PM Refresh





At the end of the Plan Phase we have different situation depending on the methodology used for managing the project

In the case of Predictive/Classic project management we have a full set of planning documents :

- Product requirement Matrix (Product Scope)
- Decomposition of all the work to perform un hierachical structure WBS WorkBreakdown Structure this is highlighting deliverables of the project
- Then all the activities to perform actual work for the COMPREHENSIVE PROJECT are organised in a Baseline Planning (i.e. estimated; organized depending of ressource availability , précédences ...)
- We have also planned for RESSOURCES , COST and RISKS for the project

AGILE Methodology at the end of the planning phase will give the following

- Product backlog splitted into Sprints and Realeases
- A first comparative estimation of complexity of users stories
- And in the case where we consider that SPRINT PLANNING meeting was performed for first sprint , then we have the Sprint Backlog (U.S. in this sprint) and the decomposition into activities for this sprint in a scrum board associated with the team resources .

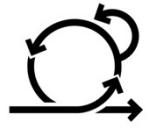
À la fin de la phase de planification, nous avons une situation différente selon la méthode utilisée pour la gestion du projet

Dans le cas de la gestion de projet Prédictive/classique, nous disposons d'un ensemble complet de documents de planification :

- Matrice des exigences Produit (Product Scope)
- Décomposition de tout le travail à faire dans une structure hiérarchique WBS (Work Breakdown Structure) qui met en évidence les livrables du projet.
- Ensuite, l'ensemble des activités pour réaliser effectivement le travail pour le PROJET COMPLET sont organisées dans une Baseline Planning (Planning initial ou de référence) (c'est-à-dire les tâches estimées; organisées suivant la disponibilité ressource , les précédences ...)
- Nous avons également planifié des RESSOURCES, COST et RISKS pour le projet dans des baselines (ou Référence initiale).

La méthodologie AGILE pour sa part à la fin de la phase de planification produira les

- Le Product Backlog découpé dans Sprints et Realease
- Une première estimation comparative de la complexité des Users Stories
- Et dans le cas où nous considérons que la réunion SPRINT PLANNING a été effectuée pour le premier sprint , alors nous avons le Sprint Backlog (Les User stories de ce sprint) et la décomposition dans les activités pour ce sprint dans un Scrum Board associé aux ressources de l'équipe .



Exécution en Méthode AGILE

Nous commençons par la méthode AGILE pour laquelle le contrôle de l'exécution est déjà « Codé » dans la méthode.

Notamment les principales réunions de suivi sont déjà décrites ainsi que leur contenu.



Making a Sprint

Réaliser un Sprint

The module will be interested in the realization of a Sprint in Agile method
It should be noted that the method presented is particularly well suited for software development but not applicable to all contexts.

Unlike the Predictive method where everything is planned in advance , here we are interested in the execution of the only slice of sprint work

The idea is therefore:

- Taking a calendar slice of the project team's activities (i.e. the one planned for the sprint),
- Planning in detail for these activities,
- Closely tracking the execution as it unfolds,
- Delivering the results by getting acceptance of the "achieved"
- Examining possible non-compliance or changes
- Learning the lessons of the execution of the installment for the rest of the project .

We will use here the follow-up recommended in the Agile / Scrum method because it is formalized by clearly identified events.

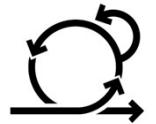
*Le module va s'intéresser à la réalisation d'un Sprint en méthode Agile
On notera que la méthode présentée est particulièrement bien adaptée pour le*

développement logiciel.

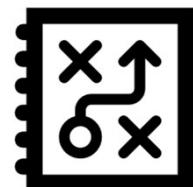
Contrairement à la méthode traditionnelle où tout est planifié à l'avance ici on s'intéresse à l'exécution de la seule tranche de travaux du sprint. L'idée est donc

- *de prendre une tranche calendaire d'activités de l'équipe projet (i.e. celle prévue pour le sprint),*
- *de planifier dans le détail ces activités,*
- *de faire un suivi rapproché de l'exécution lors de son déroulement,*
- *de procéder à une livraison des résultats en validant le « réalisé » et en examinant les éventuels non-conformités ou changements*
- *De tirer les leçons de l'exécution de la tranche pour la suite du projet .*

On va utiliser ici le suivi préconisé dans la méthode Agile / Scrum car il est formalisé par des événements clairement identifiés.

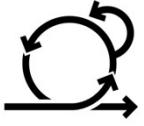


Démarrer le **SPRINT** : **SPRINT PLANNING**



The sprint Planning is fully described in module 3 of this training Course it ends up with a Sprint Backlog with the user stories that are part of the sprint and a Scrum Board with the activities that needs to be held by whom and when

Type of KANBAN follow up in Trello



Monitoring Sprint Execution: « Daily » SCRUM



The team performs the Sprint , the Scrum method provides a daily meeting called Daily Scrum during the execution.

For teams that have a part-time work schedule (parallel schooling), one may decide to make a meeting with less frequently (every 2 days for example). The meeting lasts about 15 minutes. It is only a matter of

- What did I realize yesterday
- What am I going to achieve today
- What are the difficulties

The Scrum Master can facilitate these meetings and update a KANBAN (possible support in Trello, GITHUB Project for example)

Depending on the initial schedule by distributing the Load of the Sprint one may be in different situations:

- Perfect "nominal"
- "in advance" see if we can add something in the Sprint Backlog with the Product Owner
- "late" see if we can differ a / user stories with the Product Owner or catch up
- "Crash" incident that blocks -> 'sprint freeze' -> 'resolution' -> restart of the Sprint please review the situation with customer and the Product Owner ..

We can use the estimation of the user stories in points, see in days to estimate the

realized at the time of the sprint and the rest to do. We can see if the progress is consistent. Customer relations are useful during execution if technical or functional issues arise during the Sprint run or if one is in a position to manage a situation (advance, delay, crash).

L'équipe réalise le Sprint , la méthode Scrum prévoit un **point quotidien appelé Daily Scrum** durant l'exécution. Pour des équipes qui ont un rythme de travail à temps partiel (scolarité en parallèle), on peut décider de faire un point moins fréquemment (tous les 2 jours par exemple).

La réunion est d'une durée de 15 minutes environ.

On traite uniquement de

- Qu'ai-je réalisé hier
- Que vais-je réaliser aujourd'hui
- Quelles sont les difficultés rencontrées

Le Scrum Master peut animer ces réunions et mettre à jour un KANBAN (support possible dans Trello , GITHUB Project par exemple)

En fonction du planning initial prévu en répartissant la charge du Sprint on peut être dans différentes situations:

- « nominal » parfait
- "en avance" voir si on peut ajouter qq chose avec le Product Owner
- "en retard" voir si on peut repousser une / des user stories avec le Product Owner ou rattraper le retard
- "Crash" incident qui bloque -> gel du sprint -> résolution -> restart du Sprint inclure le client et le Product Owner ..

On peut s'aider de l'estimation des user stories en points , voir en jours pour estimer le réalisé au moment du sprint et le reste à faire. On voit si l'avancement est cohérent.

Les relations avec le client sont utiles durant l'exécution si des questions techniques ou fonctionnelles se posent pendant l'exécution du Sprint ou bien si on est en situation de gérer une situation (avance, retard, crash).

The image displays two identical screenshots of a Wooclap poll interface, one above the other. Both screens show a list of five numbered options, each with a progress bar at 0% and a count of 0 people.

Allez sur wooclap.com et utilisez le code GMAQIR

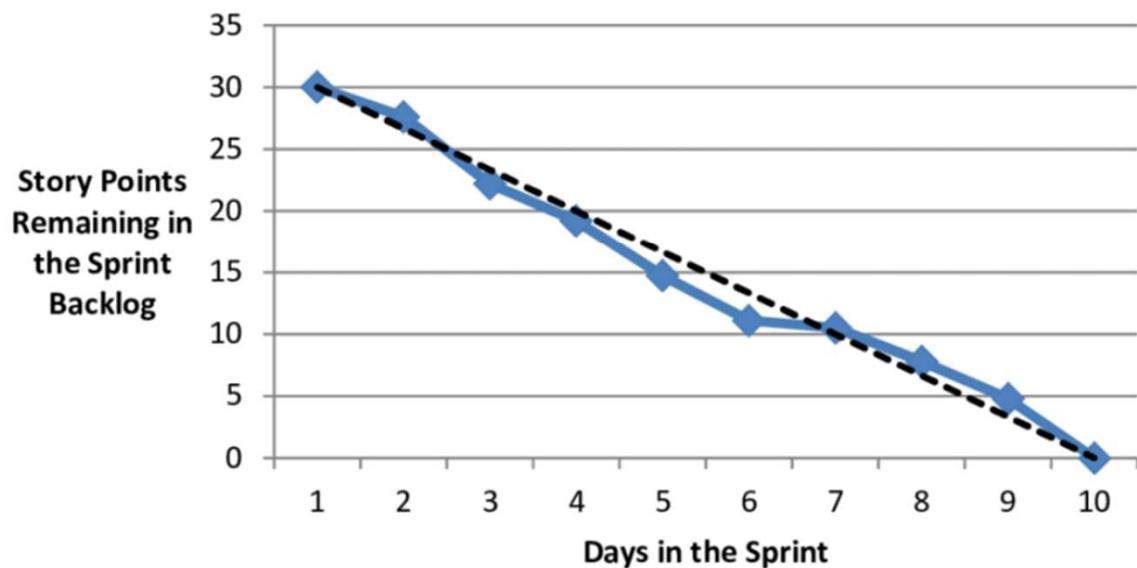
Nos sprints sont de 4 semaines : à la fin de la semaine 2, nous nous rendons compte que nous sommes très en retard, et que nous aurons beaucoup de mal à finir le contenu du sprint dans les 4 semaines ...

- 1 Aller voir le Product Owner pour demander d'allonger la durée du Sprint d'une semaine.
- 2 Décider au sein de l'équipe de dev, la user story que nous ne réaliserons pas et poursuivre nos travaux.
- 3 Solliciter le Product Owner **Cliquez sur l'écran projeté pour lancer la question**
- 4 Simplifier la réalisation de certaines User stories pour avoir moins de travail à réaliser.
- 5 Faire un développement rapide avec un risque d'augmenter la dette technique

wooclap

0 / 0

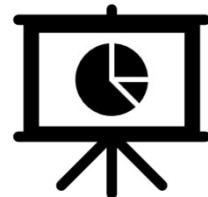
Sprint Burn-Down Chart



Just a presentation of graphical representation of Sprint burndown Chart



Fin du SPRINT : **SPRINT REVIEW**



The end of the Sprint is achieved by a meeting that includes the client: Product Owner and beneficiaries of the project called SPRINT REVIEW:

- Sprint Goal Recall (What Was Expected)
- DEMO to customer of what has been achieved
- Collect feedback and hopefully confirm the acceptance of what is produced,
- Review of difficulties with the customer,
- Review of the "VELOCITY" (was the Scoring estimate for a Sprint correct too much or not enough)
- Eventually schedule a Product Backlog Grooming meeting and/or identify the following sprint.

La fin du Sprint est réalisé par une réunion incluant le client : Product Owner et bénéficiaires du projet appelée SPRINT REVIEW :

- *Rappel du But du Sprint (ce que l'on avait prévu)*
- *DEMO client de ce qui a été réalisé*
- *Recueillir les feedback et on l'espère la confirmation du "Terminé",*
- *Revue des difficultés avec le client ,*
- *Revue de la "VELOCITE" (est-ce que l'estimation Scoring pour un Sprint était correct trop ou pas assez)*
- *Eventuellement planifier une réunion de Product Backlog Grooming et/ou identifier le sprint suivant .*



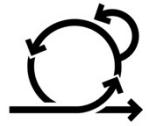
Product Backlog Grooming

Following a sprint, for example, a product owner-initiated meeting may be held for Product Backlog grooming activities:

- Clean up user stories: for example, remove unnecessary stories,
- Rephrase user stories
- Refining users stories
- Reshape sprint/release content

Suite à un sprint par exemple , une réunion à l'initiative du Product Owner peut avoir lieu pour des activités de Product Backlog grooming :

- *Nettoyer les user stories : par exemple supprimer des stories inutiles,*
- *Reformuler des user stories*
- *Raffiner des users stories*
- *Reshaper le contenu de sprint/release*



Fin de SPRINT: **SPRINT RETROSPECTIVE**



To improve the functioning of the dev team in view of the past sprint, Scrum Master organize a meeting of the dev team at the end of the Sprint Review : We are trying there to determine for example:

- What works well
- What doesn't work that well
- What remains mysterious;
- What would work better if...

Practical conclusions are drawn for the operation of the team

Pour améliorer le fonctionnement de l'équipe de dev au vu du sprint passé, réunion de l'équipe de dev à l'issue du Sprint et de la review : On cherche à déterminer par exemple:

- ce qui fonctionne bien ;
- ce qui fonctionne moins bien ;
- ce qui reste mystérieux ;
- ce qui fonctionnerait bien si ...

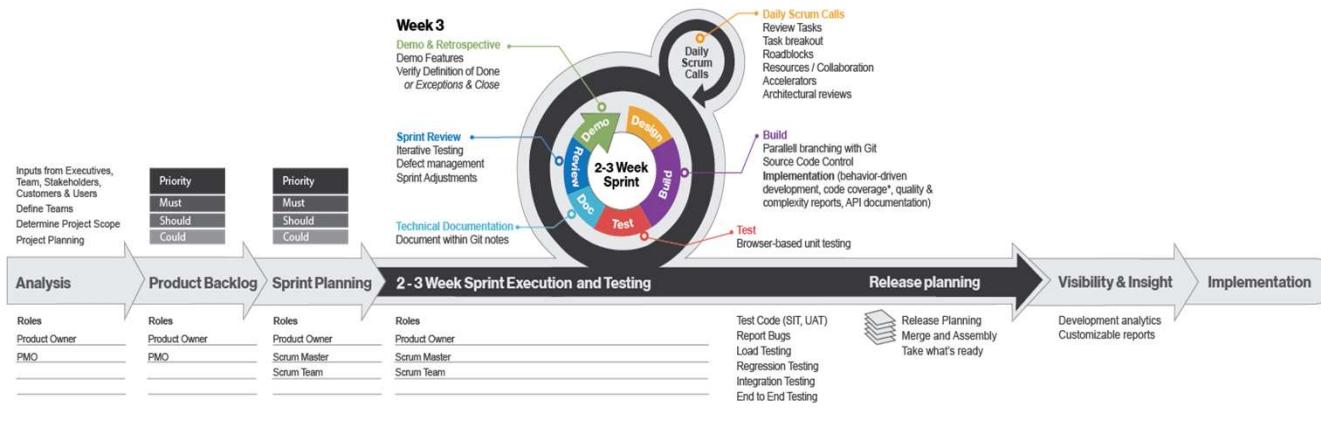
On en tire des conclusions pratiques pour le fonctionnement de l'équipe

Communication Client-Developpeurs :

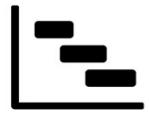
Construite dans la méthode SCRUM, si le client a opté pour AGILE, il doit impérativement se plier a minima aux évènements prévus dans la méthode.

Bien s'accorder sur les moyens (canaux, convocations, calendrier) pour fluidifier le processus.

Comprehensive view



This shows a comprehensive view of Agile method in dev. Applications (from a



Exécution en mode Prédictif/Classic

This section will focus on Predictive / Classic PM method .



BASELINES TO CONTROL

- **PRODUCT SCOPE BASELINE**
- **PROJECT SCOPE BASELINE**
- **PLANNING BASE LINE**
- **RESOURCE BASELINE**
- **COST BASELINE**
- **RISK BASELINE**

Ask students what are these baselines

PRODUCT SCOPE BASELINE = Product requirement Matrix approved by Customer at project Start

PROJECT SCOPE BASELINE = WBS and DELIVERABLES plan agreed at project start after planning

PLANNING BASE LINE = for internal to dev. Team Initial GANTT with all activities ; for discussion with Customer may be a resumé of the GANTT or Simple TIMELINE

RESOURCE BASELINE = Initial Plan for Resources supporting the PLANNING BASELINE

COST BASELINE = initial Plan of the COST spending along time associated with planning and Resource baseline.

RISK BASELINE = initial risk plan including action associated with risk



En mode prédictif, on n'a qu'à suivre le Plan, mais il va falloir faire face à de nombreux évènements imprévus....

Managing a project in predictive mode should be simple, everything has been planned in advance. So if everything goes according to plan, team just have to follow the plan.

But everyone knows that many unexpected events shall happen so we will have to develop our plans to deal with these events . For instances :

- Human factor (competence, behavior, health, etc.)
- Poor estimation of the effective work to perform
- Technical problem of all kinds
- Delay in expected supplies for customer or external providers
- External factor (strikes, change products, change in legislation ..)
- Change in customer needs
- Customer business priorities moving affecting part or all the project

La gestion d'un projet en mode prédictive devrait être simple , tout a été planifié à l'avance. Ainsi si tout se déroule normalement , il suffit de suivre le plan. Mais chacun sait que beaucoup d'évènements imprévus vont se produire qu'ils faudra donc aménager nos plans pour y faire face .

Par exemple :

- *Facteur humain (compétence, comportement, santé...)*
- *Mauvaise estimation des tâches*
- *Problème techniques de tout ordres*
- *Retard des fournitures attendues externes au projet*

- *Facteur externes: (grèves, changement produits, changement législation ..)*
- *Variation du besoin client*
- *Changement de cap du client, remise en cause de tout ou partie*



Mettre en place un cadre de communication pour contrôler le déroulement du projet

Rather than a theoretical presentation of process of managing and controlling the execution phase of the Project, we propose to review main communications events thru the project and illustrate then what are the actual actions that are undertaken as a result of these events.

Exemple de plan de communication

Type d'évènement	But	Fréquence habituelle
Réunion de lancement	Se faire rencontrer les acteurs Partager le plan projet Fixer clairement les outils /moyens de communication utilisés	au démarrage du projet
Revue interne Equipe	Voir l'exécution des travaux en cours (équivalent Daily Scrum AGILE)	2 à 3 fois par semaine
Revue Client / Comité de Pilotage	Voir l'exécution et les fournitures de part et d'autre Valider des réalisations intermédiaires (Equivalent Sprint Review Agile)	Hebdomadaire le plus souvent
Processus de gestion des changements	Processus se déroulant lors des comité clients pour les changements de périmètre (équivalent au AGILE PB Grooming)	En fonction des demandes
Comité Directeur	Grande Validation, Orientation, Gestion de crises	Trimestriel ou selon la demande

Formalisez le plan de communication :

Faites le approuver par le client,

Planifier la logistique des évènements de com : Placer les RDV, Format et diffusion des Comptes rendus

Kick Off Meeting : Lancement



Exemple d'outils :

- Envt Technique : Products, DevOps, Test tools, physical env...
- Repository : GitHub, GitLab..
- Suivi d'activités : **Trello** , **GITHUB Project**, JIRA...
- Communication : Mail, Slack, Skype, Discord...

During Project kickoff the tools shall be explicit and initialized with correct parametrizing .
Here are some example for software dev. Type of project



Internal review meetings

Réunions de suivi internes

These meetings allow to review activity with the team, the pace of these meetings is minimum weekly and often more frequent. The frequency can be adapted to the situation of the project. Frequently this rhythm will become everyday in a crisis situation or tense situation. For example a delivery that must take place in 10 days while there is still a lot of work to be completed, This would justify a daily meeting to perfectly synchronize participants, and take corrective actions as soon as possible. These meetings are used to verify what has been done, what will be undertaken for the next period. It allows us to talk about the problems we have encountered. Possibly updating activities in the tracking tool.

This is an opportunity for the project manager to check whether alignment with the plan remains valid, and if not to analyze the reasons why, and see if a plan to return to the original schedule is possible.

If this is not possible, the PM should analyze the impacts modifying the plan accordingly and inform stakeholders.

A schedule to date is then built with visibility on the deviations from the baseline.

Ces réunions permettent de faire le point avec l'équipe , le rythme de ces réunions est minimum hebdomadaire et souvent plus fréquent. Le rythme peut être adapté à la situation du projet . Fréquemment ce rythme deviendra journalier en situation de crise ou situation tendue par exemple une livraison qui doit impérativement intervenir dans 10 jours alors qu'il reste beaucoup de travail à achever va justifier un point journalier

pour parfaitement synchroniser les participants , prendre les mesures correctives au plus vite .

Ces réunions permettent de vérifier ce qui a été fait, ce qui va être entrepris pour la période suivante . Elle permet d'évoquer les problèmes rencontrés. Possiblement mise à jour des activités dans l'outil de suivi.

C'est l'occasion pour le chef de projet de vérifier si l'alignement avec le plan reste valide, et dans le cas contraire d'analyser les raisons , voir si un plan de retour au planning initial est envisageable .

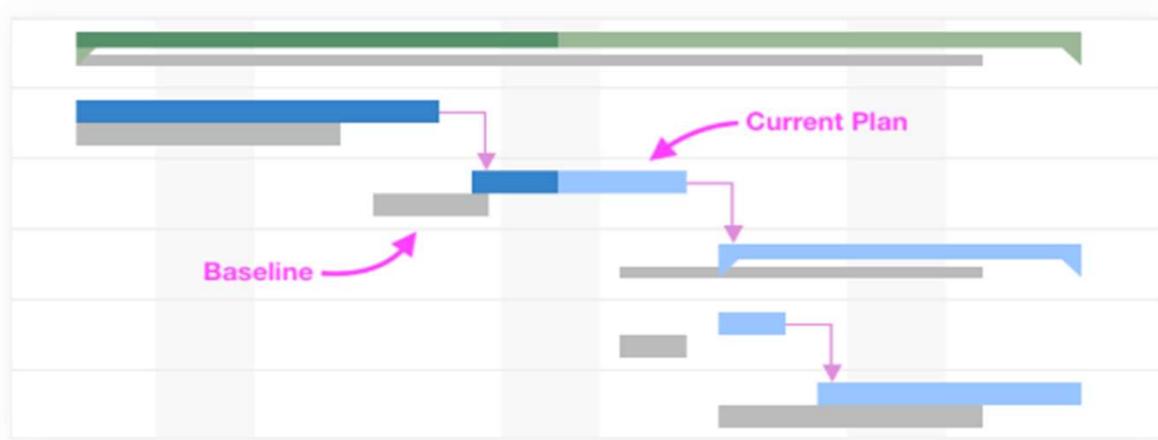
Si ce n'est pas possible , le PM doit analyser les impacts modifier les plans en conséquence et en informer , les parties prenantes.

Un planning à date est alors construit avec visibilité sur les écarts vis-à-vis de la baseline.

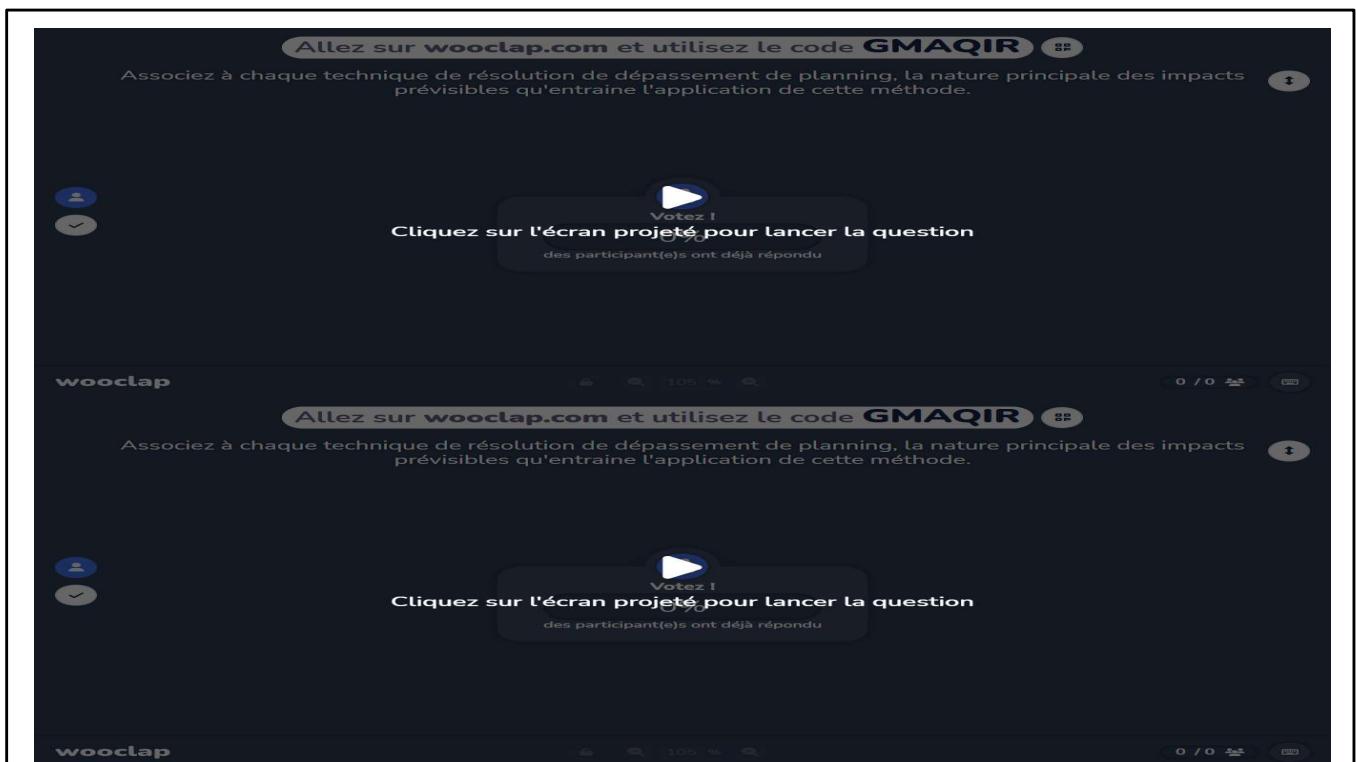
Kanban : des outils qui peuvent aussi être utilisés pour le suivi des travaux des équipes :

Mettre dans *Trello* les activités identifiées dans le Gantt et suivre leur avancement.

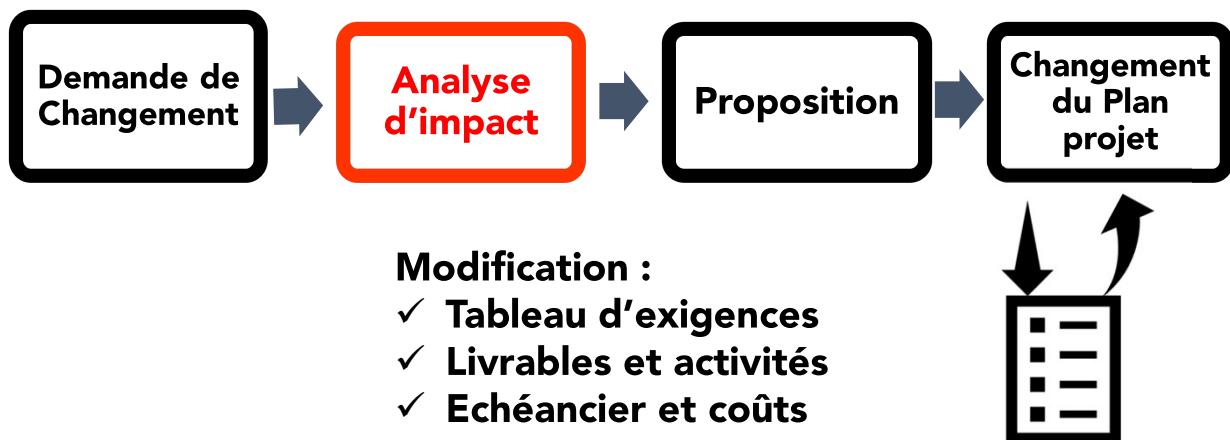
Analyse des retards : Gantt Chart



This slide demonstrate the way of measuring impact on planning of a delay in project execution.



Gestion des changements



Le principe est le suivant :

Le sous-traitant, ou le client identifient un changement par rapport au contenu du projet.

Le client établi une demande de changement.

Le prestataire effectue une analyse d'impact pour la prise en compte de ce changement et il soumet au client une proposition, qui décrit les impacts en termes de contenu, de prix et

de délai.

Si le client accepte cette proposition, elle constitue alors un Avenant au contrat en ce sens qu'elle modifie un certain nombre de ses éléments constitutifs.

Les délais de réponse à une demande de changement, et d'autres éléments de ce processus sont en général inclus dans le contrat.

C'est souvent un outil très important pour les sociétés de service, pour l'amélioration de la marge des projets : en effet, ces avenants sont rarement soumis à la pression de la concurrence, comme c'est le cas pour l'appel d'offres initial.

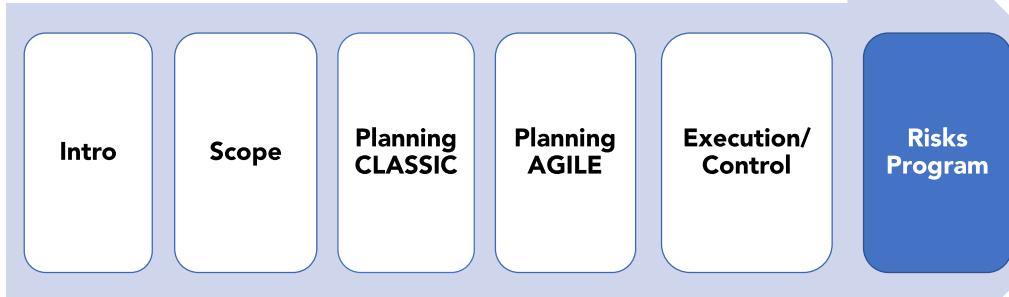
En résumé :

En méthode Agile appliquez les processus de contrôle prévus par la méthode.

En classique définir un plan et une méthode de suivi et de contrôle, insistez pour que le client y joue bien son rôle.

Réagissez vite aux imprévus, aux difficultés, pour maintenir la cohésion et la cohérence du projet en impliquant le client fortement.

ING02 – PM Refresh



Gestion des Risques !

Objectifs de ce module :

- * Savoir identifier les risques les plus importants qui pèsent sur vos projets,
- * Être capable de mesurer leur impact potentiel,
- * Connaître les principales stratégies de réponse,
- * Suivre l'évolution de ces risques lors du

déroulement du projet,

C'est une pratique qui permet de maximiser les chances de succès du projet.

C'est un des domaines clés du Project Management (PMI).

Un risque individuel du projet est un évènement ou une condition possible (incertaine) dont la concrétisation aurait un impact positif ou négatif sur un ou plusieurs objectifs du projet.

La définition d'un risque est fournie ici.

A noter que l'événement doit être **incertain**, il peut se produire ou non. S'il n'est pas incertain, ce n'est plus un risque mais une **hypothèse** ou une **contrainte** du projet.

L'impact doit être réel, donc l'occurrence du risque conduit à soit un impact financier, soit un impact planning, soit un impact sur une

des fonctions du produit.

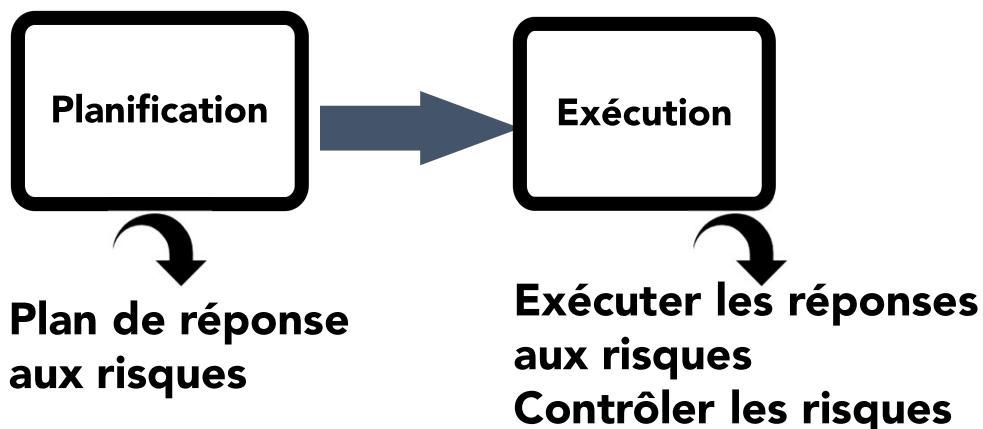
Lorsque que le risque a un impact positif on appelle cela une **opportunité**.

Le risque global du projet ou niveau de risque du projet est l'effet de l'incertitude sur l'ensemble du projet, provenant de toutes les sources d'incertitude possibles, comme les risques individuels.

Souvent le risque Global du projet sera exprimé sous la forme d'un niveau : **Faible, Moyen, Elevé, Très élevé.**

Ce niveau de risque global impliquera un niveau d'analyse plus ou moins poussé dans la gestion des risques

Gestions des Risques



La gestion des risques est un processus continu :

Lors de la phase de planification du projet, **un plan initial de réponse aux risques est établi (Risk Baseline)**

Pendant l'exécution du projet :

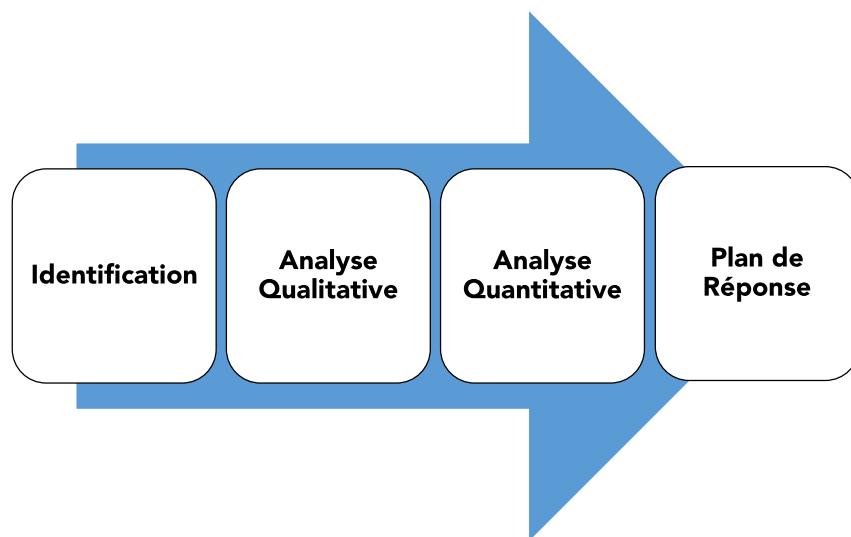
- * D'une part on applique les actions prévues dans le plan de réponse aux risques,
- * D'autre part on prend en compte l'émergence de nouveaux risques, les risques peuvent changer et certains peuvent être fermés, d'autres peuvent devenir plus probables, ou avoir un impact différent.

La gestion des risques doit être correctement **adaptée en fonction du niveau d'exposition aux risques global estimé.**

Ce niveau de risques global peut être examiné suivant plusieurs axes :

- * Taille du projet
- * Complexité du projet
- * Importance du projet
- * Approche de développement (Agile itérative ou Waterfall ?)

Gestion des risques : La méthode



La méthode d'élaboration du Plan de réponse initial (Risk Baseline) comprend 3 Processus :

- * L'identification des risques
- * L'analyse qualitative
- * L'analyse quantitative

Le plus souvent, une organisation de

développement comme une société de services proposera à ses chefs de projets un cadre (framework) incluant des fichiers de suivi, des questionnaires etc...

Il y a un risque que cet événement arrive à cause de cette situation et cela aura cet impact

Ce type de formulation est à employer quand nous allons identifier et formaliser les risques individuels.

Registre des risques : RISK log

Risk ID #	Titre du Risque	Description du risque (Il y a un Risque que X se produise à cause de Y)	Impact Description
1	<i>exigences fonctionnelles incomplètes</i>	<i>Il y a un risque de réalisation qui ne convienne pas au client à cause d'un besoin faiblement décrit</i>	<i>Retravail sur les livrables pour qu'ils soient acceptés par le client</i>

On enregistre les risques dans un registre (feuille Excel) des risques.

Ce fichier est souvent standardisé au niveau des organisations. On verra, qu'au fil de l'application de la méthode de gestion des risques, il va s'enrichir de nombreuses colonnes.

Plan de réponse



Pour déterminer le plan de réponse on identifie :

- 1/ la stratégie de réponse parmi les 4 stratégies possibles (voir slides suivantes)
- 2/ on détermine l'action à mener pour mettre en œuvre la réponse
- 3/ on planifie comme une activité du projet intégrée à celui-ci la réponse décidée (on modifie la liste des activités, l'échéancier, la

tables des couts en conséquence)

4/ si le risque n'est pas éliminé mais amoindri
on évalue à nouveau le risque résiduel

En résumé :

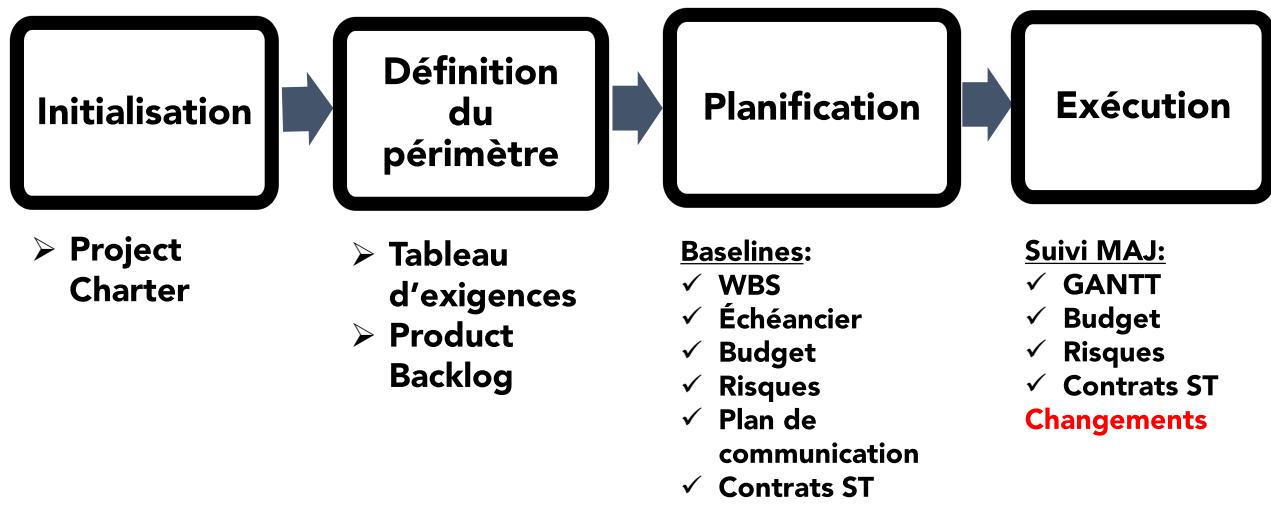
Identifiez les risques qui peuvent affecter le projet,

Filtrez pour ne conserver que les plus significatifs,

Définir des stratégies et des plans d'action pour les risques retenus,

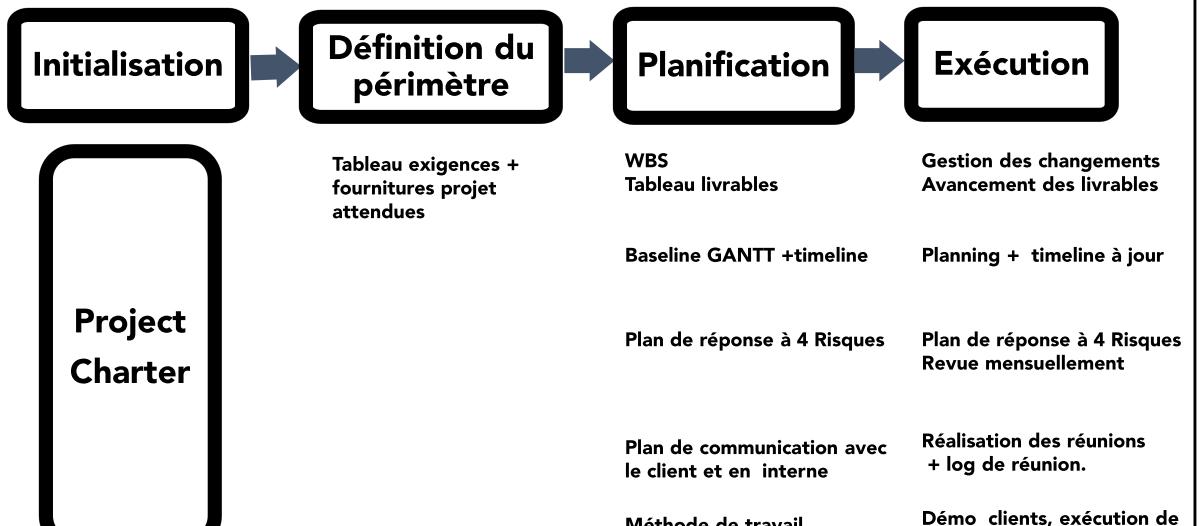
Au cours de l'exécution repassez en revue très régulièrement les risques pour éliminer ceux qui ne sont plus d'actualité, revoir les plans et identifier si de nouveaux risques apparaissent.

Récap livrables Gestion de Projet

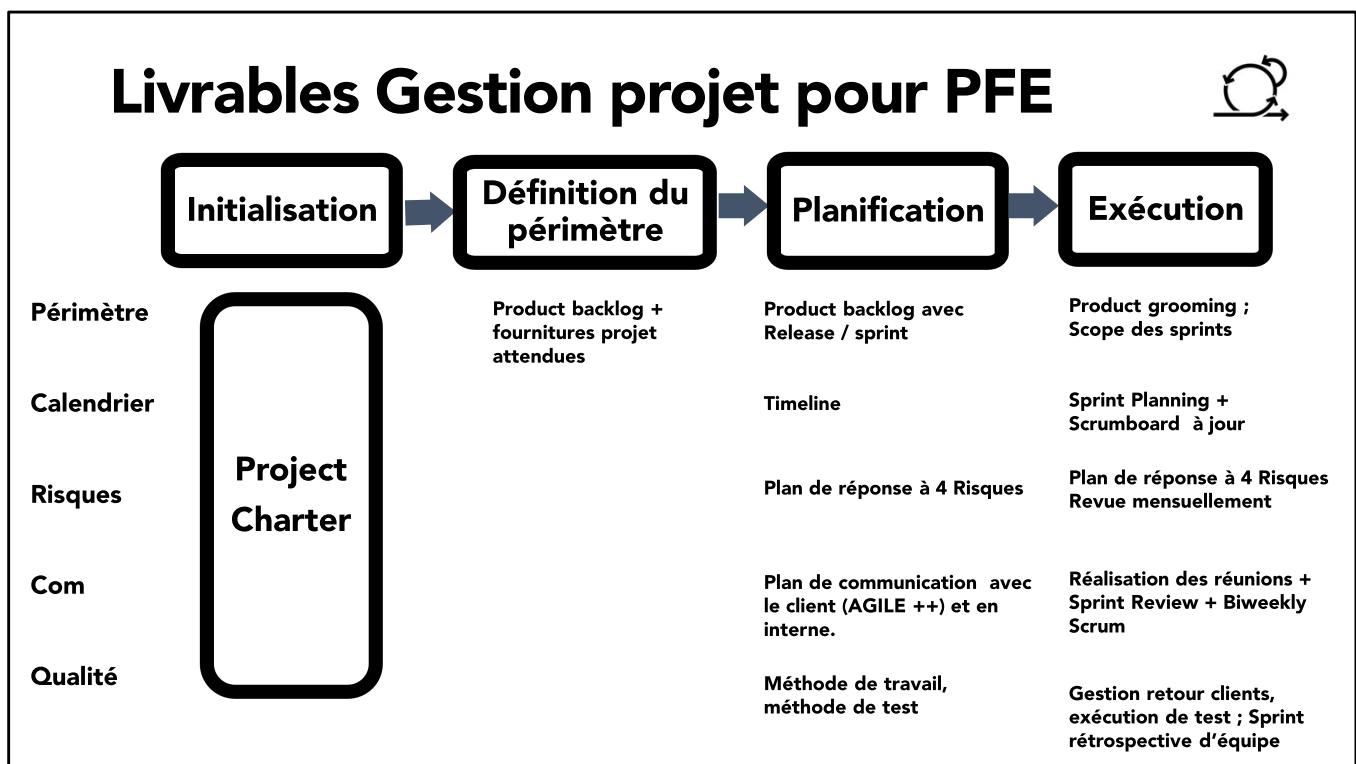


On peut rajouter à ce récapitulatif

Livrables Gestion projet pour PFE



Recapitulatif des livrables en mode classique pour un PFE



Récapitulatif des livrables en mode Agile pour un PFE

Avant de partir...

Allez sur wooclap.com et utilisez le code **GMAQIR** 

Après cette séance de Refresh...

 1 Vous êtes confiant pour utiliser la gestion de projets pour piloter vos projets de fin d'étude. 0% 0 

 2 Vous trouvez que c'est très compliqué.  0% 0 

Cliquez sur l'écran projeté pour lancer la question

 3 Vous pensez que ça ne sert à rien. 0% 0 

 4 0% 0 

wooclap   105 %  0 / 0  

