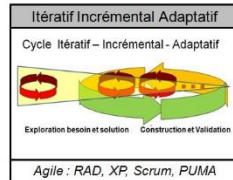
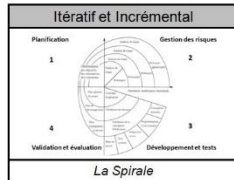
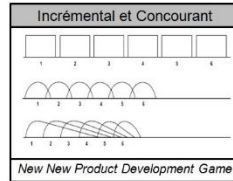
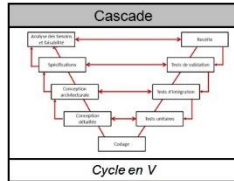


# Cycle de vie – Introduction exigences

Philippe Finkel, Philippe Cluzeau

ESIPE  
mars 2021



# Définir le temps

## **Le temps**

La construction d'un logiciel s'exprime dans le temps.

Il faut en général un effort coordonné dans le temps de plusieurs personnes pour finaliser la construction d'un logiciel.

Pour organiser le travail, il faut le planifier et donc le décrire dans le temps.

# Etape

**Étape** : une fois l'étape atteinte, il n'y a pas de retour arrière. L'étape est **définitive**, elle **jalonne** le processus de développement où il n'y pas de retour en arrière.

C'est un des objectifs du processus de gestion : **aller de l'avant mais aussi exprimer le fait que certaines choses sont irréversibles** (par exemple: la livraison d'une version au client).

# Activité et tâche

**Activité** : une activité consomme du temps, exige une compétence et a pour objectif un livrable. Elle peut être répétée si le livrable n'est pas satisfaisant.

**Tâche** : c'est le regroupement d'une activité et de l'évaluation (**validation**) du livrable associé.

# Projet – Etude

**Projet** : entreprise temporaire qui a pour but de fournir un livrable défini à l'avance.  
La particularité d'un processus de gestion de projet est d'être composé d'éléments irréversibles : les tâches.

**Étude** : moyens affectés à la recherche de solutions pour un problème donné.

# Définitions initiales

## Unité de Travail

Il faut prendre le terme unité comme **non découparable** (ex : 9 mois pour une femme enceinte). Et dans le sens **maîtrisable** : je peux m'engager sur la réalisation d'une tâche en un temps fini. Travail et évaluation sont indissociables, il faut qu'une évaluation intervienne pour terminer la tâche.

# Définitions / Cycle de vie I

Découpage du Projet

**Etape** = Phase + jalon

**Processus** : suite de phases composées de tâches

**Cycle de vie**

Le cycle de vie est une présentation générale des étapes d'un projet et de l'ordonnancement de celles-ci

# Le besoin du chef (responsable) de projet ?

- 1) Assurer la bonne terminaison du Projet
- 2) Maîtriser l'avancement
- 3) Affecter judicieusement les ressources
- 4) Anticiper les problèmes de coûts / délais / qualité
- 5) Pour répondre à cet ensemble de contraintes, la structuration et la planification sont fondamentales. Mais quelle structuration devons-nous utiliser ?



## Cycle de vie II

1) Il y a deux niveaux le projet le développement

2) Vous devez gérer les deux niveaux :

- **le niveau projet** qui part de la **demande**, d'éléments contractuels et qui aboutit à un **livrable** avec de la valeur, c'est l'aspect projet
- **le niveau logiciel** où la granularité de l'activité de base est plus petite, et où vous gérez la création logiciel à proprement parler.

# Vue projet

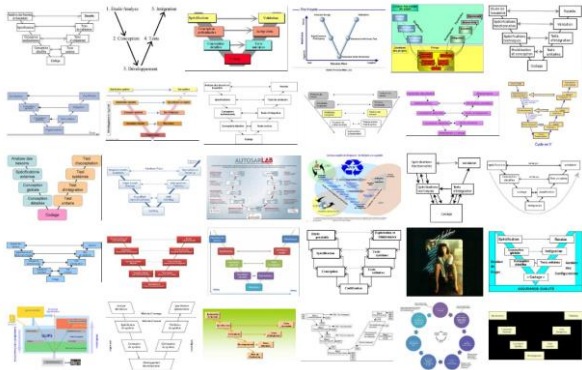


Vue projet

## Cycle de vie III

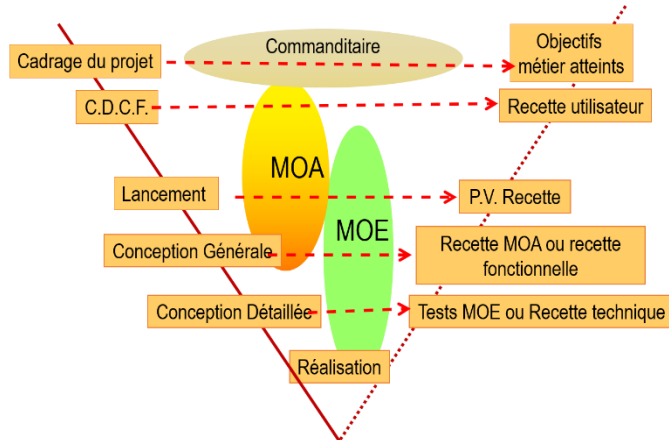
- 1) Comme vous l'avez constaté, cette approche projet ne nous donne aucun outil pour construire les éléments clefs de notre projet logiciel
- 2) Quelle architecture ?
- 3) Quels utilisateurs ?
- 4) Ni aucune assurance sur la validité des différents éléments produits dans ce cycle

# Cycle en V



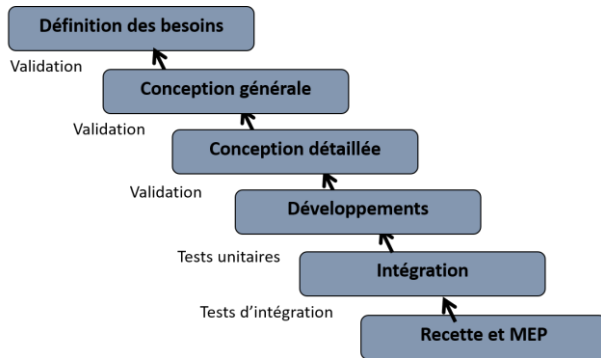
Le Cycle en V

- 1) Comme vous pouvez le voir Google Images a quelques cycles en V en stock ...
- 2) Le Cycle en V a un mérite : il associe chacune des phases de production avec une phase de validation/contrôle ce qui tombe sous le sens pour l'assurance qualité



Cycle en V en quelques mots

# Cycle en cascades – processus projets



Cycle en Cascades- processus Projet

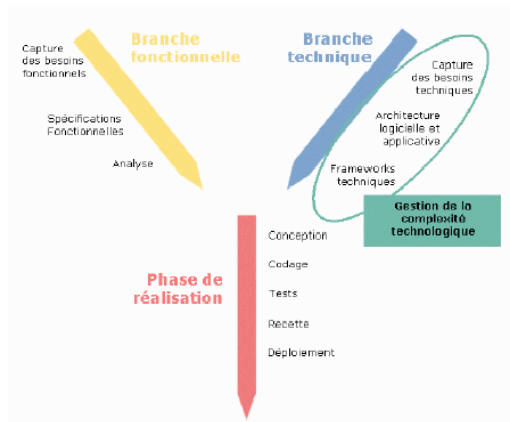
# Cycle de vie 2TUP / Y – processus projets

1) L'idée forte est que l'on a deux grandes problématiques dans un projet logiciel :

- le besoin fonctionnel, ce que doit faire le logiciel
- les contraintes techniques

2) Ces deux problématiques ne font pas intervenir les mêmes interlocuteurs chez le client et dans l'équipe projet d'où une démarche à 2 branches :

- L'une pour gérer la capture des besoins fonctionnels
- L'autre pour gérer la capture des besoins techniques

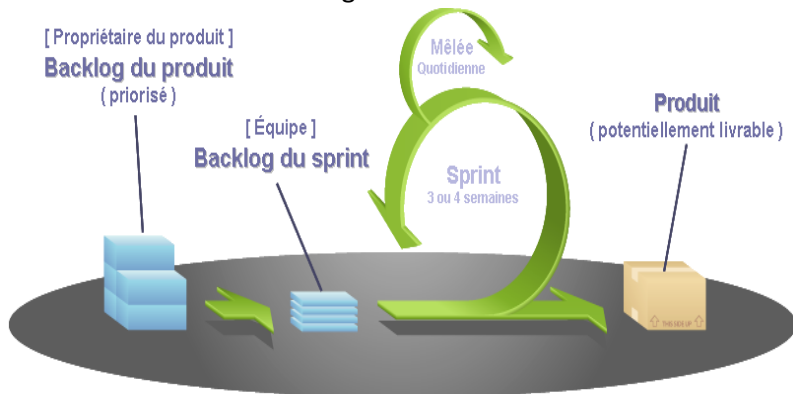




# Cycle de vie itératif

L'idée de base d'un cycle de vie itératif est de livrer au plus tôt quelque chose qui puisse être vu / testé / utilisable par le client et procéder ensuite par affinage / modifications successifs.

## Modèles agiles – Scrum



# Choisir un cycle de vie

Le choix du cycle s'apprécie en fonction du projet considéré :

## 1) **Un développement purement technique :**

Une grosse étude technique est obligatoire ainsi qu'un besoin de technicité de la part de l'équipe projet embauchée.

Si les spécifications sont complètes => une approche en cascade / V

Si les spécifications ne sont pas claires => étude -> cascade / V

## 2) **Un développement Additionnel :**

La Part technique a déjà été écrite : il suffit de la réutiliser (relire et comprendre bien sûr)

La part fonctionnelle est nouvelle car le développement additionnel doit répondre à de nouveaux besoins.

Les spécifications sont complètes -> une approche en cascade

Spécifications pas claires => ? Le Client est Présent => SCRUM, spiral etc. Pas de client => ECHEC DU PROJET

## 3) **Très Haut niveau des Exigences :**

Ex: vies humaines en jeux, très longue durée d'exploitation, production "One shot".

Une Modélisation Formelle des Spécifications et gestion des Exigences fonctionnelles et techniques

Une gestion de la Qualité doit être réalisée en parallèle de la production et de la gestion de projet.

-> Cycle 2TUP

# Gestion de projet (Wikipedia) - Etapes

L'approche par **étapes** (« **jalons** » voir Jalonnement) est un acte de direction, qui permet de bien structurer le projet dans le temps, en y apportant de nombreuses garanties pour le maître d'œuvre : sa progression est calendairement mieux suivie.

- Lancement
- Expression des Besoins/Spécifications
- Faisabilité
- Analyse/Développement
- Réalisation/Vérification/livraison
- Exploitation

# Lancement

**1)Phase préliminaire** : la réflexion sur l'intérêt du projet en lui-même, en termes **d'opportunité** stratégique.

**2)Jalon de lancement du projet** : on décide (au niveau « politique ») qu'il y a lieu de lancer un projet spécifique, et on y consacre un chef de projet, une équipe, des moyens, un responsable et un budget.

# Expression des besoins / spécifications

**1)Phase d'expression du besoin** : la définition de ce que l'on attend (les fonctions attendues), le périmètre, ce sur quoi on va évaluer le projet, ce qui est important et ce qui l'est moins.

**2)Jalon de validation du besoin** : le client valide l'expression de ses besoins. Ce sont les bases sur lesquelles le projet va être bâti.

# Faisabilité

**1)Phase de faisabilité** : l'étude de ce qui est techniquement et économiquement faisable. Consultation des maîtres d'œuvres possibles, comparaison des propositions techniques et financières des réalisateurs possibles.

**2)Jalon du choix de la solution** : signature du contrat qui précise ce qui sera fait et la manière de le faire.

# Analyse

- 1) **Analyse** : le maître d'œuvre coordonne les travaux sur le “produit papier”, pour préciser ce qui doit être fait jusqu’au dernier boulon.
- 2) **Jalon de lancement du chantier (éventuel)** : quand le “produit papier” est suffisamment défini, on peut faire le point avant de lancer les travaux de réalisation.

# Réalisation / Vérification / Livraison

- 1) Phase de réalisation** : le chantier est lancé, les travaux avancent pour transférer le “produit papier” dans le réel.
- 2) Phase de vérification (qui peut commencer très tôt, sur le “produit papier”)** : sur le produit réel ou sur le produit papier, on vérifie (ou on calcule) que les caractéristiques attendues sont bien au rendez-vous (avec les écarts éventuels, qu’il faut alors gérer).
- 3) Jalon de livraison (et recette) encore appelée acceptation** : on remet le produit entre les mains du client, qui en devient propriétaire (et peut émettre des réserves sur les écarts constatés). C’est la fin du projet proprement dit.



# Exploitation / Jalons

**La phase d'exploitation**, qui commence le plus souvent par la levée des réserves constitue la fin de la relation contractuelle.

**A retenir :**

**Les jalons** permettent de faire le point sur le projet et de n'engager la phase suivante que si tout va bien.

Les décisions actées lors de cette revue de changement de phase sont des éléments stables sur lesquelles peut être bâtie la suite du projet.

Le jalonnement se préoccupe moins du contenu de chaque phase, que de l'appréciation de son résultat, où le client (ou maître d'ouvrage) est amené à se prononcer.

# Les phases d'un projet logiciel

- 1) Inception
- 2) Ecriture des exigences
- 3) Ecriture des spécifications
- 4) Réalisation Analyse Conception Intégration Tests
- 5) Vérification Recette
- 6) Déploiement
- 7) Validation Contrôle

# Inception

- 1)l'avant projet
- 2)Demande
- 3)Budget
- 4)Planning
- 5)Ressources
- 6)Acteur

En fonction, c'est la Direction qui décide de répondre à la demande

# Exigences

**Gestion des exigences :** activité qui consiste à exprimer les exigences

Une exigence (projet ou produit) représente la description de ce que le projet ou le produit doit « faire ». C'est le QUOI, la substantifique moelle du CDC !

# Ecriture des spécifications

**Ecriture des spécifications** : activité qui consiste à transformer les besoins du client en spécifications utilisables afin de réaliser l'ingénierie du produit à concevoir.

A l'issue de cette activité, on doit aboutir à des exigences testables.

## Phase

DEBUT: Commence à partir du moment où on a les exigences

FIN: quand les spécifications sont testables

## Acteurs

Les spécifications sont construites par un dialogue avec le demandeur

## Sous-activités

- Dessiner l'interface graphique
- Modéliser
- Prototyper
- Un projet d'étude
- Glossaire

# Analyse

**Analyse** : activité qui consiste à formuler une solution au problème sous une forme abstraite (modèle d'architecture, modèle d'analyse...)

Architecture Technique

Architecture Fonctionnelle

Architecture globale

Problématique d'intégration contextuelle

**début fin**

Début : Commence avec les Exigences

Fin : le modèle d'architecture est stabilisé ; le plan d'intégration est écrit et validé.

**Acteurs**

Architecte Exploitation MOA délégué

# Conception

**Conception** : activité qui consiste à formuler une solution concrète en terme de technologie d'implantation à partir de la forme abstraite de la solution.

## **Phase**

DEBUT: L'Ensemble des documents des phases précédentes sont livrés

FIN: L'ensemble des doutes sur la faisabilité pratique sont levés

La liste des LOTS et leur ordonnancement est faite

## **Acteurs**

Equipe de développement

## **Sous activités**

Ecriture des Diagrammes UML, de classe et de séquence

Glossaire

Sous - Planification de la réalisation

# Réalisation

**Réalisation** : activité qui consiste à transformer des actifs projet conceptuels (langage ou modèle) en des actifs exécutables.

**Début fin**

DEBUT: Fin de la conception

FIN: Tous les lots sont terminés et conformes aux spécifications

**Acteurs**

Développeurs



# Intégration

**Intégration** : activité qui consiste à assembler progressivement par morceau le produit final afin de s'assurer que celui-ci fonctionne selon les attentes.

**Début fin**

Début : l'intégration commence avec chaque fin de LOT de développement

FIN: Quand l'ensemble des lots sont intégrés dans le logiciel final

**Acteurs**

Intégrateur

# Tests / Vérification / Contrôle

**Tests** : activité conduite pour évaluer la qualité de l'actif logiciel afin de l'améliorer par identification de défauts et de problèmes.

**Vérification/Contrôle :**

activité qui consiste à s'assurer que le produit fabriqué respecte les spécifications (modèle formel dérivé des exigences).

**Validation / recette :**

activité qui consiste à s'assurer que le produit fabriqué respecte les exigences négociées avec le client.

# Déploiement

Le déploiement doit parfois être vue comme un projet à part entière

Il faut bien sur avoir un logiciel opérationnel pour se lancer dans le déploiement

**Déploiement** : activité qui consiste à porter les actifs projet sur la cible hôte.

**Problèmes de déploiement** : gestion des configurations

**Acteurs**

C'est une activité spécifique idéalement confiée à des profils techniques chevronnés

**Pilotage** : activité qui consiste à s'assurer entre autre que les jalons sont respectés, gestion des ressources, anticipation des dérives, etc.

# Organisation

- 1) Construire le projet est le travail du chef de projet
- 2) Le chef de projet doit définir le cycle de vie (la mise en phases)
- 3) Affecter les ressources
- 4) Calculer le coût

# Quelques définitions avant le TD ! (1/2)

## Glossaire

Le glossaire est un document ou une partie de document définissant les concepts que manipule le système et les termes métier clés du projet.

Un terme doit être renseigné si :

- Le terme / l'expression est spécifique à un domaine métier ;
- Il y a plusieurs définitions possibles d'un même mot ;
- La définition métier au sein de l'Organisation diffère de celle communément admise ;
- Il y a un risque de malentendu ou d'incompréhension lié à l'utilisation d'un terme.

Quand doit-on initialiser et compléter le glossaire ?

Celui-ci doit être initialisé dès le tout début du projet.

Le glossaire est un , il est alimenté à n'importe quelle étape du projet. C'est un document vivant !

- Les définitions doivent les plus courtes possible, claires et non ambiguës
- Les acronymes doivent être explicités dans leur version longue, mais également définis clairement.

Le glossaire est un excellent outils pour écrire le cahier des charges car les utilisateurs et le client reconnaissent les concepts et ils les utilisent pour décrire ce que doit faire le système.

Ce modèle des Objets du Domaine est la base de nombreuses approches objets, où il est en suite injecté directement sous forme de classes dans l'application.

# Quelques définitions avant le TD ! (2/2)

## Acteurs

Un acteur est une entité qui définit le rôle joué par un utilisateur ou par un système qui interagit avec le système modélisé. Ils peuvent être humains, non humains ou indirects.

En UML :

- les acteurs apparaissent dans les diagrammes de cas d'utilisation.
- Les acteurs humains sont des utilisateurs du logiciel à travers son IHM par ex.

## Parties prenantes :

Une partie prenante (ou stakeholder en anglais) est une personne, un groupe de personnes ou une organisation qui impacte ou peut être impacté par le projet. Elle peut affecter ou être affectée directement ou indirectement, de façon positive ou négative, par un ou plusieurs aspects du projet. Elle a donc des intérêts dans le projet et a une influence sur la prise de décision. Il est important d'être attentif aux attentes et aux besoins des parties prenantes pour garantir la réussite d'un projet.

## Frontières / périmètre

Le périmètre de projet définit les frontières du projet du point de vue des parties prenantes et couvre la totalité des livrables inclus dans le projet. Il peut être modifié au fil du projet ( Revue de Projet – Nouveaux choix du Commanditaire ) et il importe donc de spécifier dès le démarrage du projet ce qui sera et ne sera pas dans le périmètre du projet.