

# Méthodes de conception orientées objets et agiles

Introduction

Découpage d'un projet

Cycles de vie logiciel

Le processus unifié.

# Processus logiciel

- Définition

Un processus définit une séquence d'étapes, en partie ordonnées, qui concourent à l'obtention d'un système logiciel ou à l'évolution d'un système existant.

Objectifs

Produire des logiciels de qualité qui répondent aux exigences des utilisateurs dans des délais et des coûts déterminés

- Deux axes de contrôle:

- Axe de développement => Contrôler la qualité du logiciel

- Axe de gestion => contrôler les coûts et les délais

# Maîtrise d'ouvrage et maîtrise d'oeuvre

- Maîtrise d'ouvrage (MOA)
  - Le maître d'ouvrage est le demandeur du livrable
  - Représenté par un responsable de projet ou par une direction de projet.
  - Le maître d'ouvrage peut faire appel à un intermédiaire pour l'expression des besoins du projet appelé maître d'ouvrage délégué.
- Le maître d'œuvre (MOE) doit réaliser le livrable demandé selon les conditions (délai, qualité et coûts) définies dans le contrat.
  - La MOE doit désigner un chef de projet (pour la conduite du projet)
  - La MOE peut faire appel à des sous traitant.

# Découpage d'un projet

- Découpage du projet: le projet est décomposé en plusieurs étapes
  - Caractéristiques d'une étape
    - Chaque étape conduit à un résultat défini
    - Chaque étape nécessite des ressources définies
    - Contraintes d'enchaînement entre étapes (séquentiel, parallèle)
    - Une étape peut être décomposée en sous étapes
  - Le découpage peut être réalisé selon deux dimensions: temporelle (succession d'étapes, de phases et de tâches) ou structurelle (modularisation).

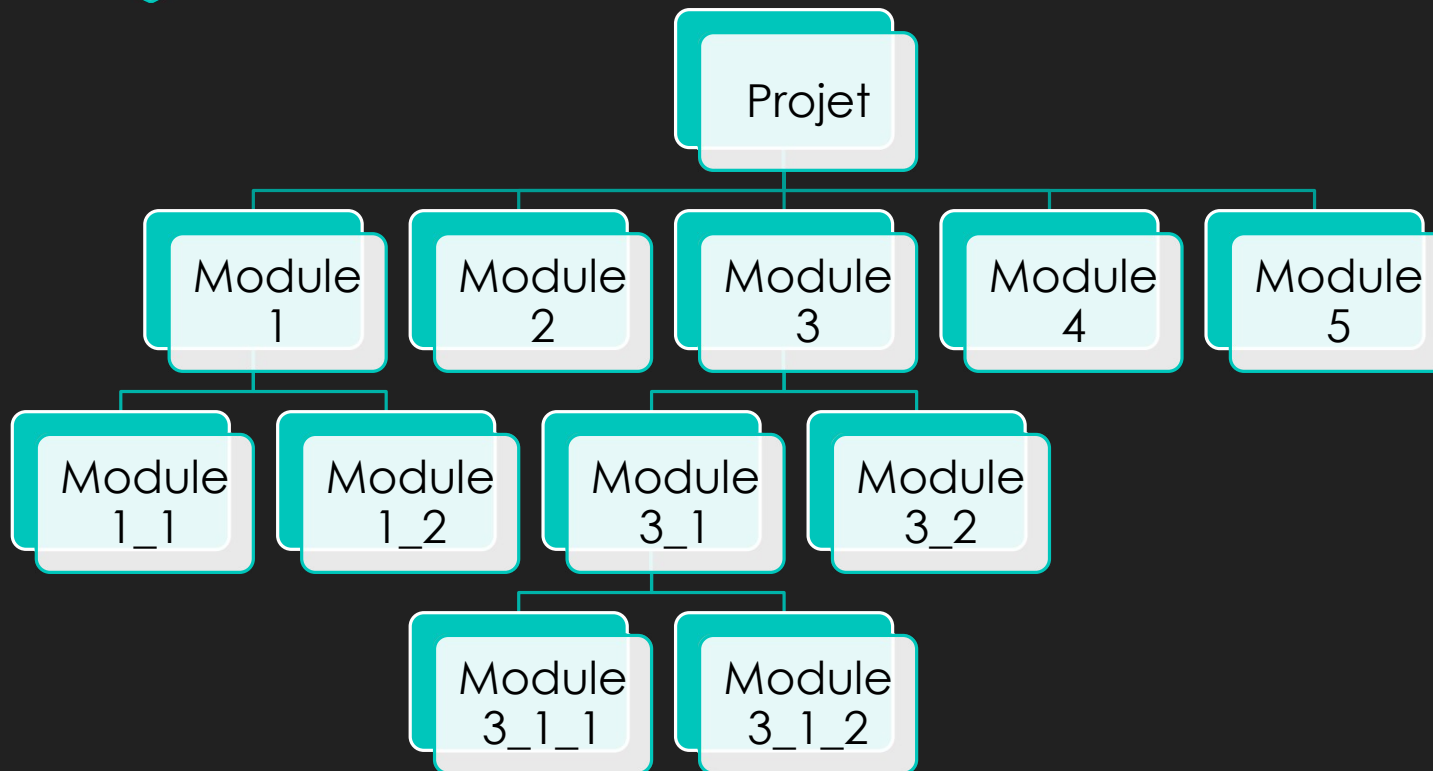
# Objectifs du découpage

- Découpage structurel
  - Réduire la complexité du système
- Découpage temporel
  - Définition de jalons intermédiaires permettant de valider la conformité du résultat de l'étape par rapport aux exigences
  - Maîtriser les risques, les coûts, et les délais.

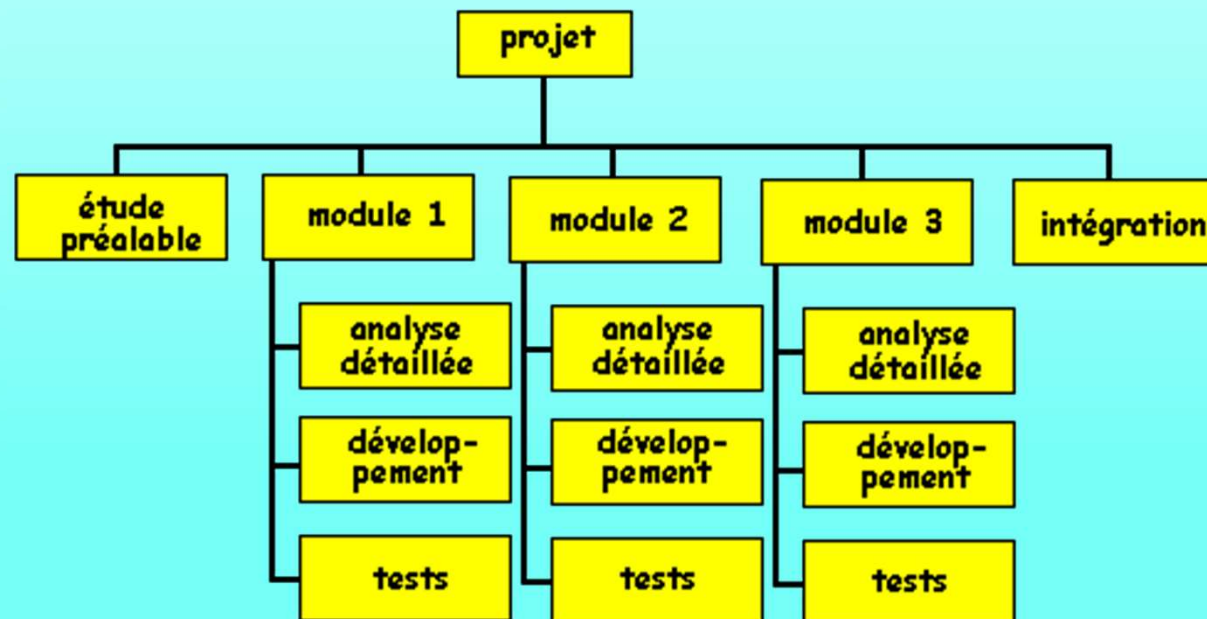
# Méthodes

- Découpage structurel
  - PBS (Product Breakdown Structure)
    - Décomposition structurelle du produit
  - Work Breakdown Structure
    - Décomposition structurelle des tâches
  - OBS (Organizational Breakdown Structure): affectation d'un responsable pour chaque unité

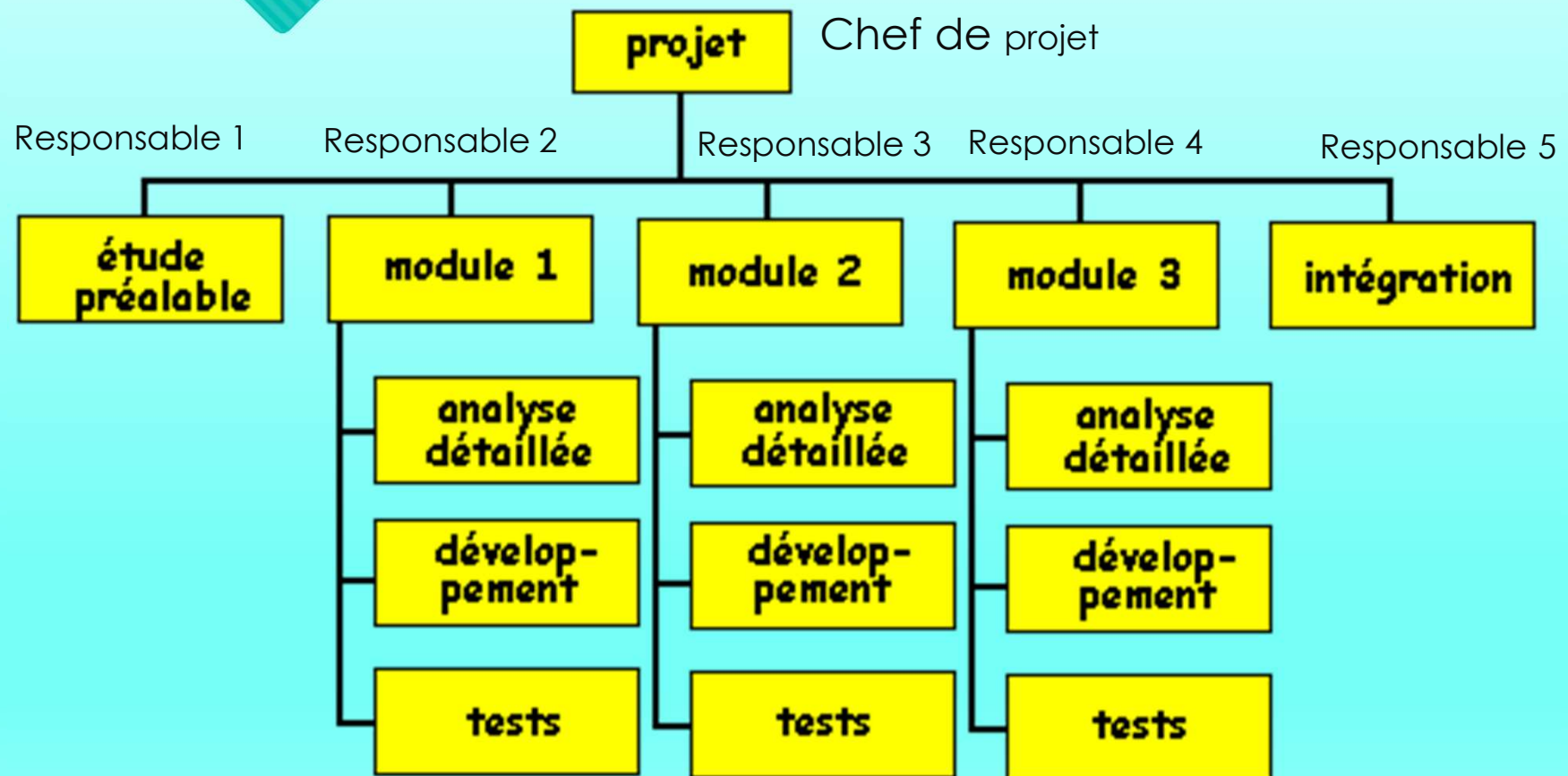
# PBS



# WBS



# OBS



# Découpage Temporel: Approche par jalons

Etude de faisabilité

Choix d'une solution

phases  
préliminaires

phases  
d'études

phases  
de réalisation

phases  
post production

avant-projet

expression des besoins

expression détaillée des besoins

analyse fonctionnelle

étude technique

lancement

développement

recette

mise en production

maintenance

Jalon de lancement du projet

Jalon de validation des besoins

production d'un document.

cahier des charges fonctionnel

cahier des charges techniques

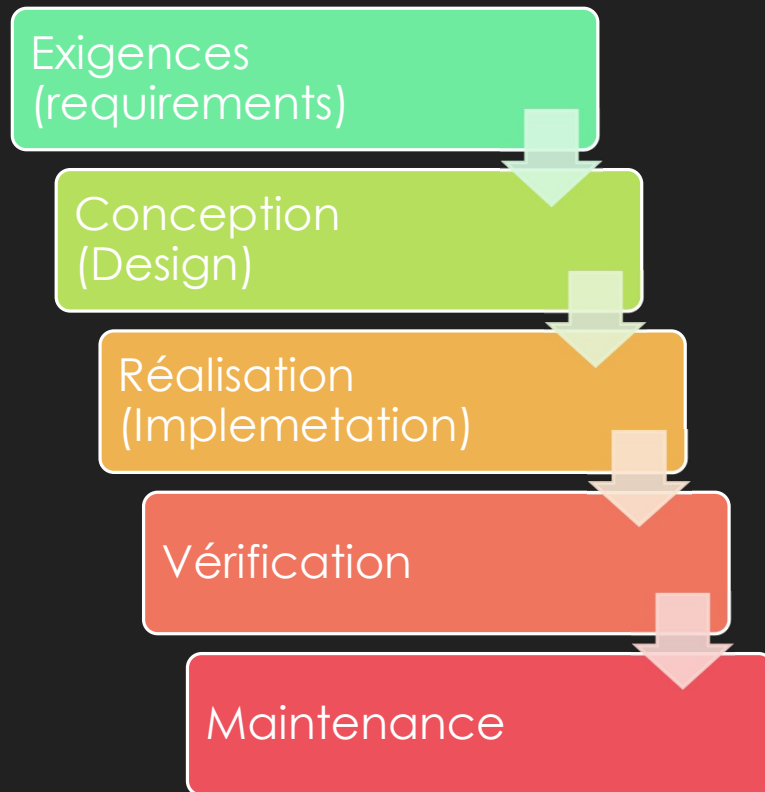
documentation projet

Jalons

# Cycles de vie logiciels

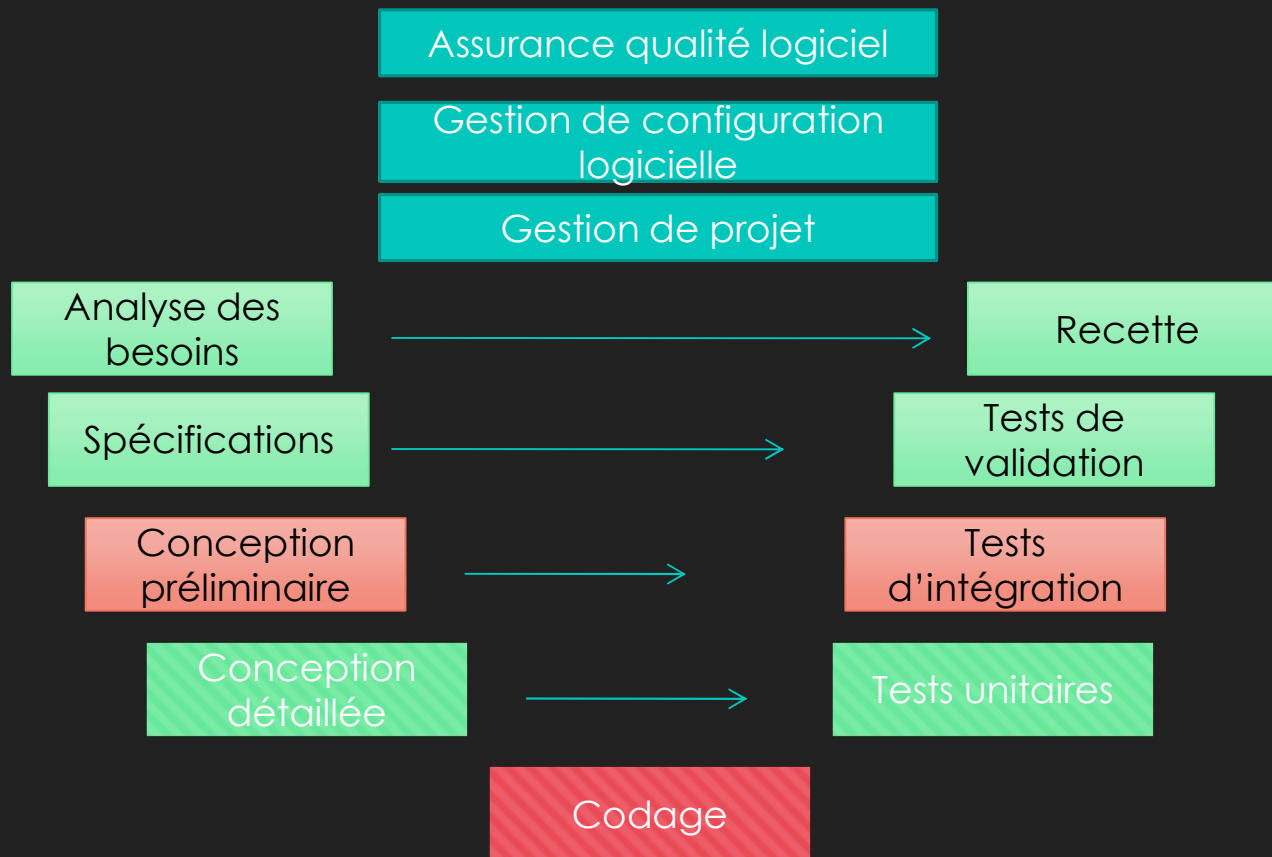
- Définition: Description d'un processus modélisant les étapes de l'élaboration d'un livrable logiciel depuis l'expression des besoins jusqu'à la fin de vie du produit.
- Outils
  - Atlassian Jira
  - Microsoft Azure Devops

# cycle en cascade



- Le plus ancien (hérité du bâtiment)
- Toute étape doit être achevée avant de passer à la suivante
- Toute modification en amont du cycle a un impact majeur sur les coûts en aval.

# Cycle en V



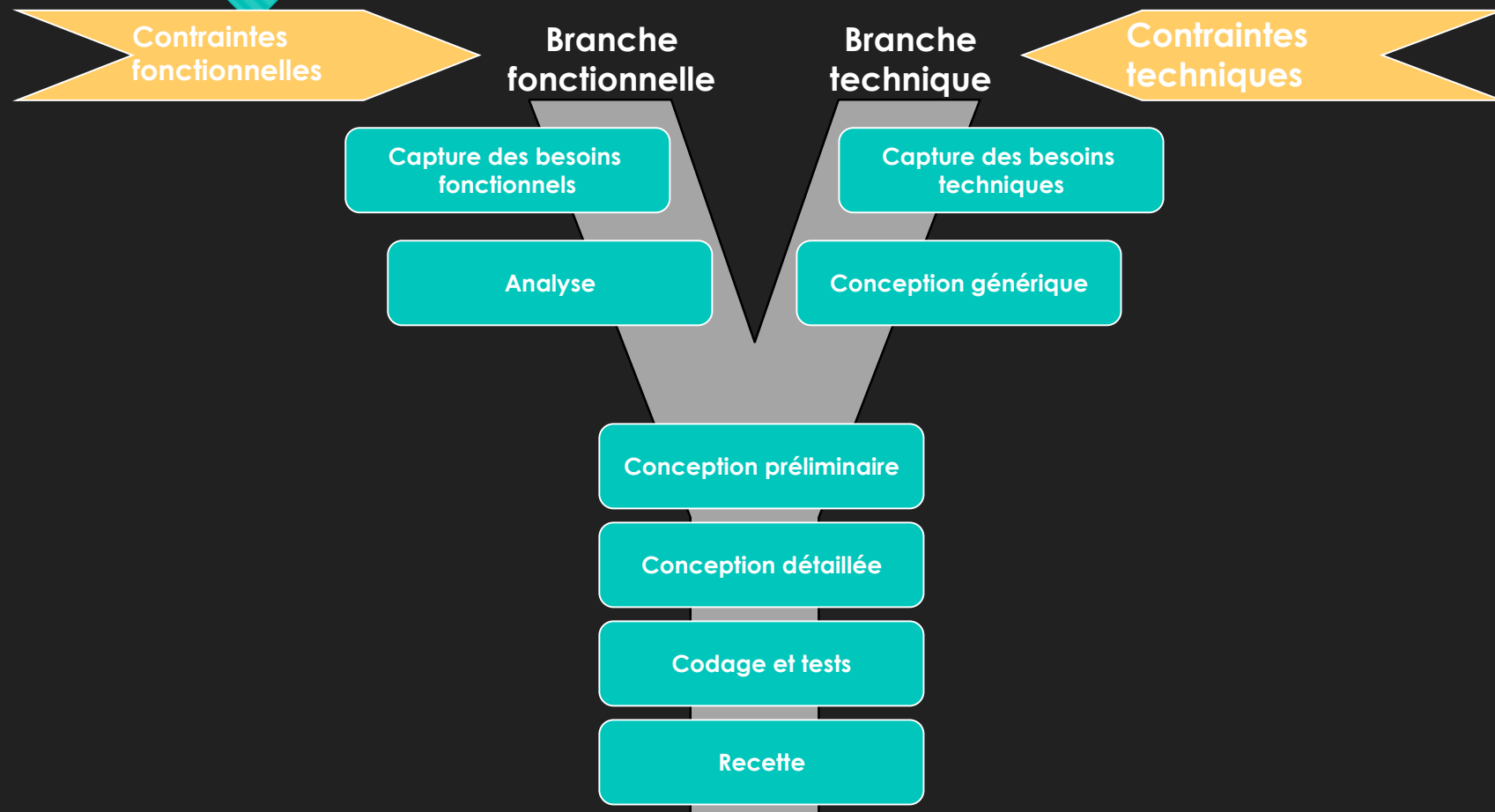
## Inconvénients :

- I1) l'utilisateur, ou client intervenant uniquement au niveau 1, doit attendre les tests de validation (appelés aussi tests d'acceptation ou de qualification) pour s'assurer que ses exigences ou besoins ont été pris en compte de manière satisfaisante

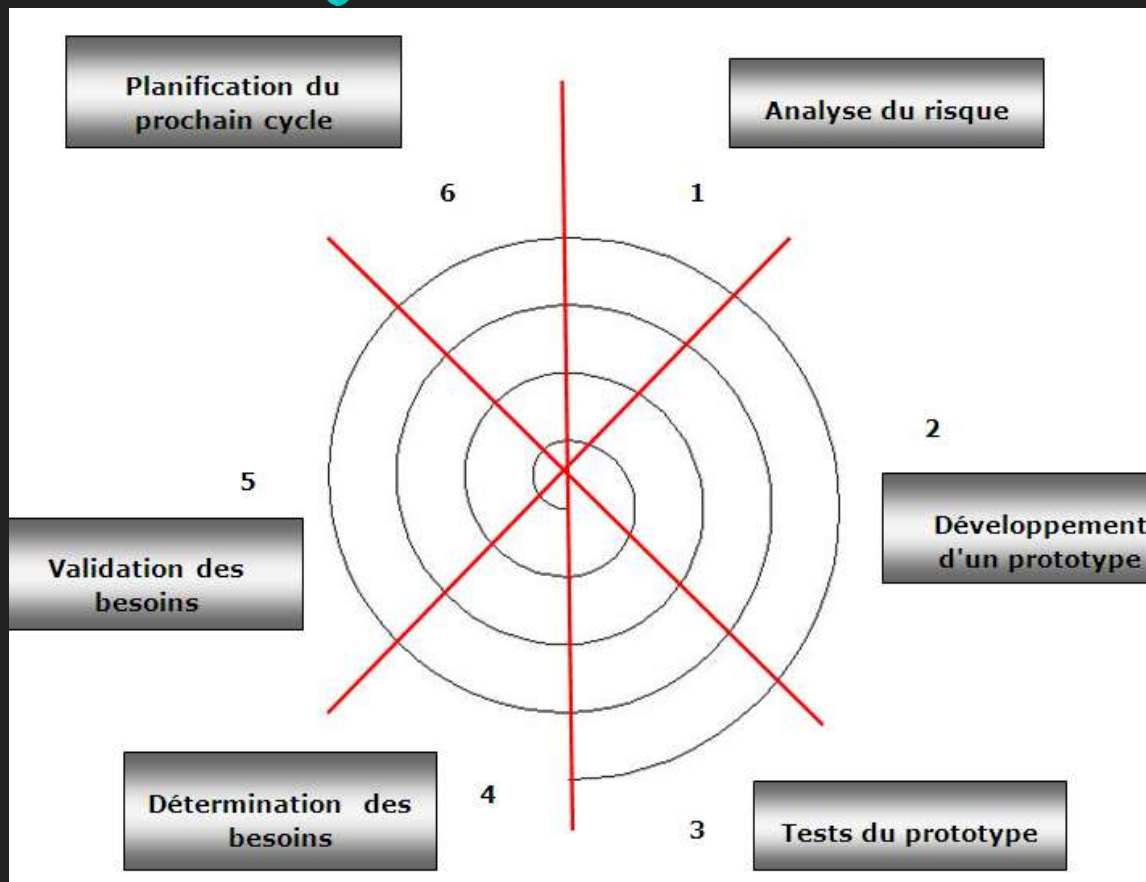
## Avantages :

- A1) modèle normalisé
- A2) montre l'interaction non seulement entre les phases successives mais aussi entre les phases de même niveau
- A3) le cycle de vie du logiciel est intégré dans le cycle de vie du système

# Cycle en Y



# Cycle en spirale



- Modèle itératif, donc chaque cycle produit une version opérationnelle du logiciel.
- Introduit la notion d'analyse de risques

# Les méthodes itératives et agiles

- Le manifeste agile en 2001 (mis au point par 17 experts)
  - 4 valeurs fondatrices
    - Individus et interactions plutôt que processus et outils
    - Développement logiciel plutôt que documentation exhaustive
    - Collaboration avec le client plutôt que négociation contractuelle
    - Ouverture au changement plutôt que suivi d'un plan rigide

# Les 12 principes de l'agilité

- première priorité: satisfaire le client en livrant tôt et régulièrement des logiciels utiles.
- Le changement est tolérable, même dans les étapes avancées du développement
- Livraisons fréquentes d'applications fonctionnelles (2 semaines à deux mois)
- Artistes et développeurs doivent collaborer quotidiennement au projet.
- Bâissez le projet autour de personnes motivées. croyez en leur capacité à faire le travail.
- La méthode la plus efficace de transmettre l'information est une conversation en face à face.
- Un logiciel fonctionnel est la meilleure unité de mesure de la progression du projet.
- Les processus agiles permettent un rythme de développement soutenable..
- technique et qualité de la conception excellentes améliorent l'agilité
- Simplicité
- Equipes auto-organisées.
- Brainstorming À intervalle régulier, l'équipe réfléchit aux moyens de devenir plus efficace, puis accorde et ajuste son comportement dans ce sens.

# Méthodes itératives et incrémentales

- XP ( eXtreme Programming): Recueil de meilleurs pratiques
  - Programmation en binôme
  - Adaptée aux petites équipes
  - Conception et test continus
- SCRUM: (le terme désigne mêlée dans le rugby).
  - Scrum ne couvre aucune technique d'ingénierie du logiciel nécessite une méthode complémentaire comme XP.
- DSDM: dynamic software development method.
  - Inspirée de la méthode RAD.