

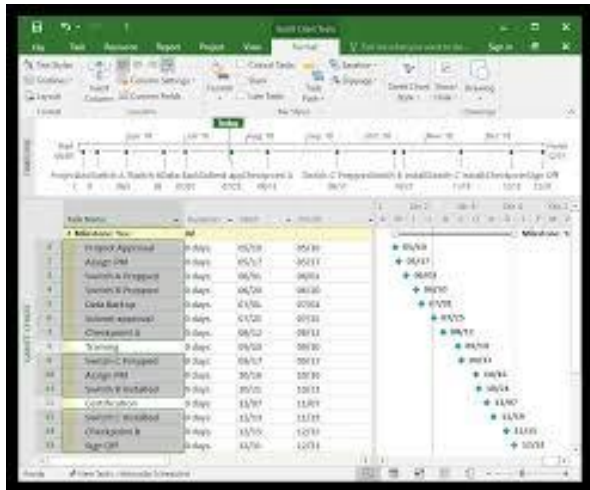
Introduction

2243.2 Génie logiciel

Définitions : logiciel

"En informatique, un logiciel est un ensemble de séquences d'instructions interprétables par une machine et d'un jeu de données nécessaires à ces opérations."

<https://fr.wikipedia.org/wiki/Logiciel>



Définitions : génie logiciel - 1

Artisanat

savoir-faire particulier, hors contexte industriel

Ingénierie

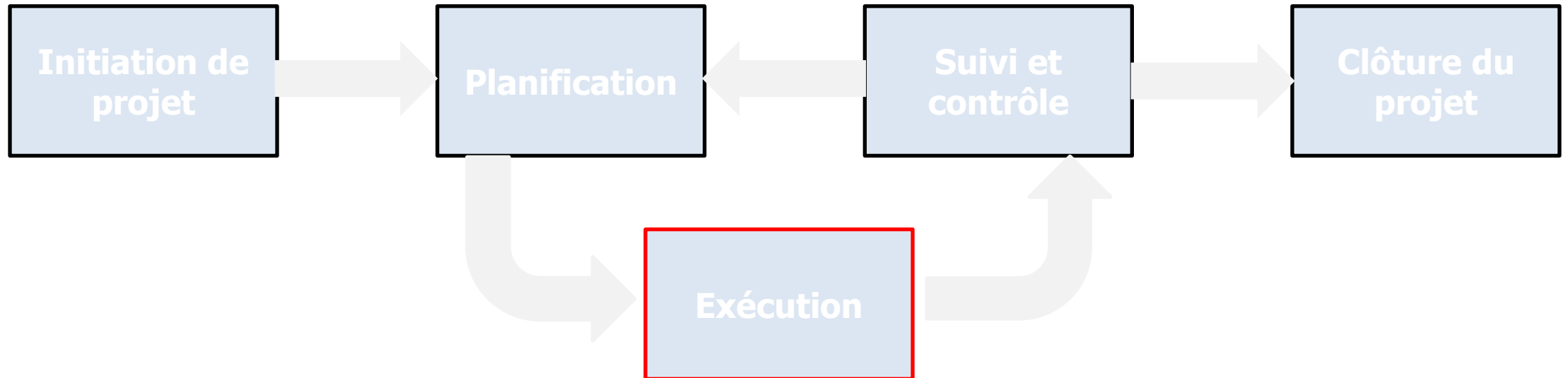
Théories, méthodes, technologies **reconnues** dans un contexte industriel

Définitions : génie logiciel - 2

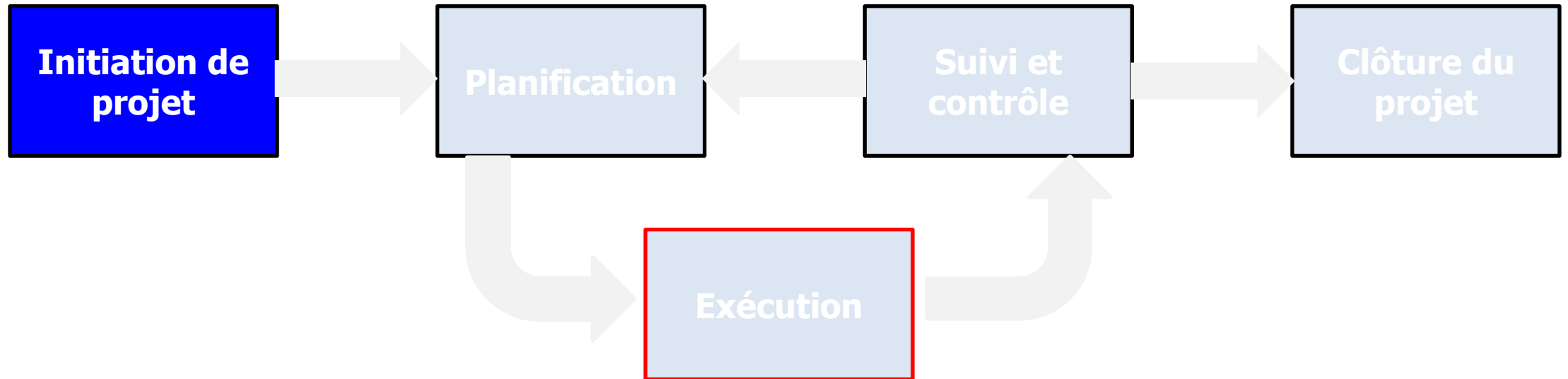
"Le génie logiciel, ou l'ingénierie logicielle, (en anglais : software engineering) est une science de génie industriel qui étudie les méthodes de travail et les bonnes pratiques des ingénieurs qui développent des logiciels."

https://fr.wikipedia.org/wiki/Génie_logiciel

Feuille de route du cours

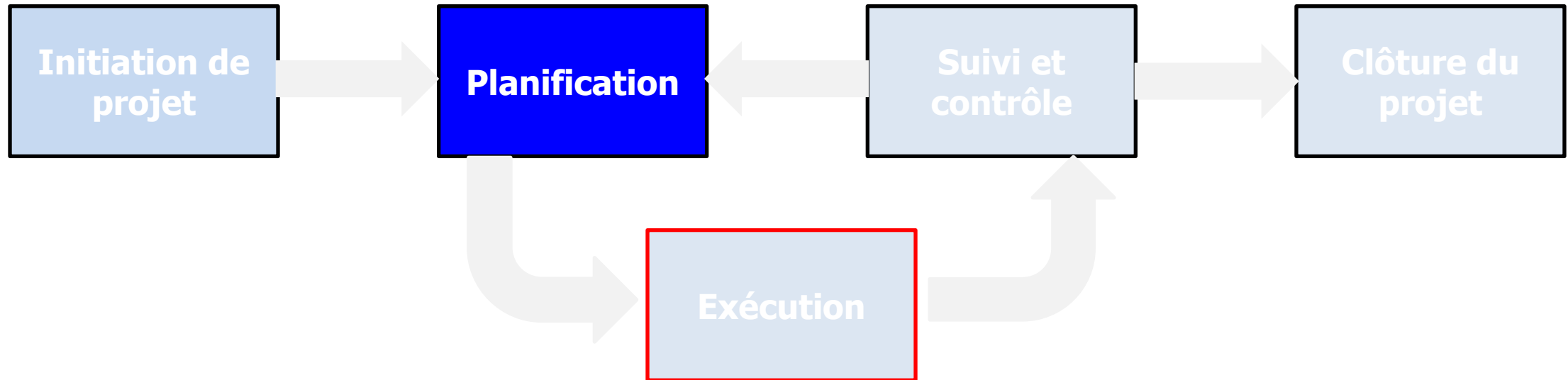


Feuille de route du cours



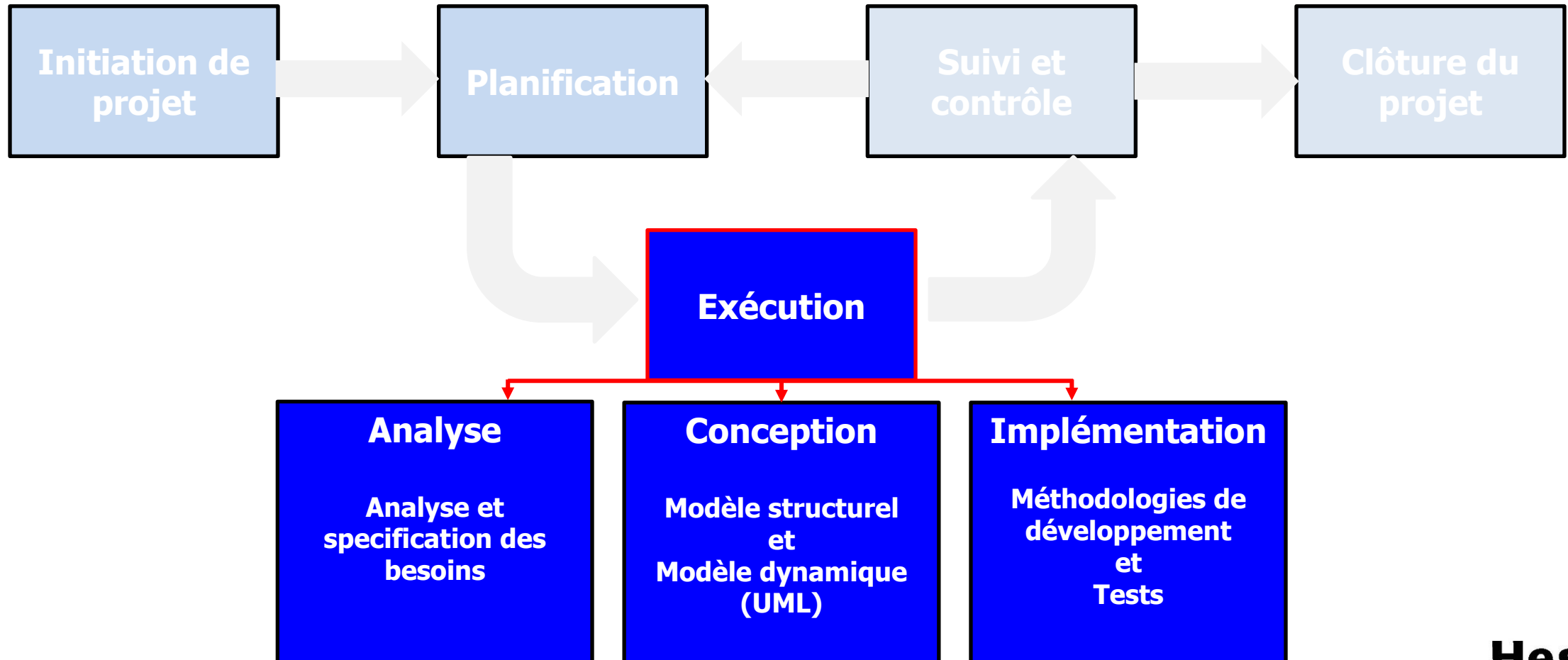
Activités, ressources, outils (IDE, support), normes, contrôles, planification de la gestion du projet

Feuille de route du cours

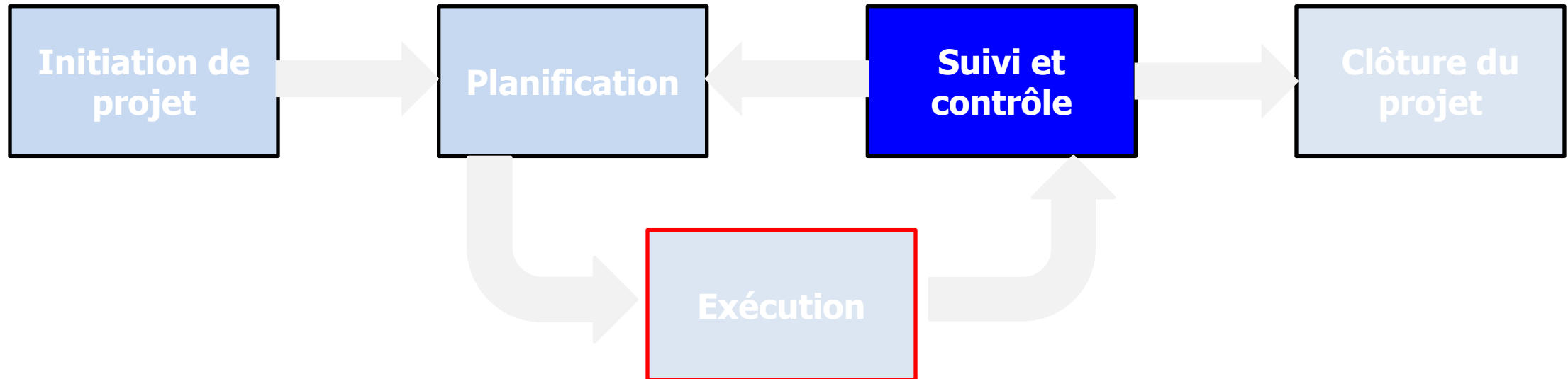


Découpage en sous-tâches, estimation du travail, durée, coûts, risques, etc

Feuille de route du cours

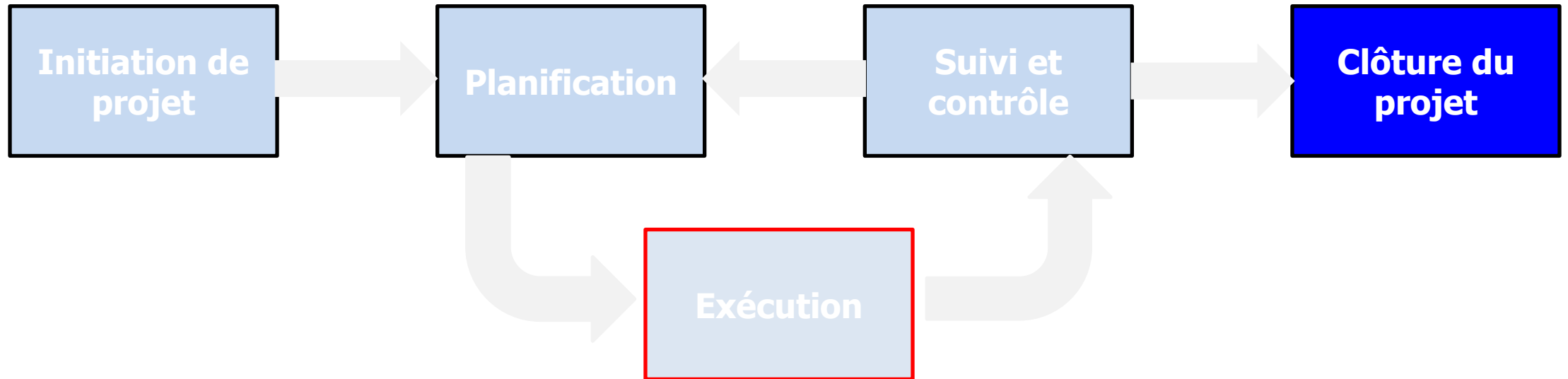


Feuille de route du cours



Avancement, délais, coûts, risques, ajustements

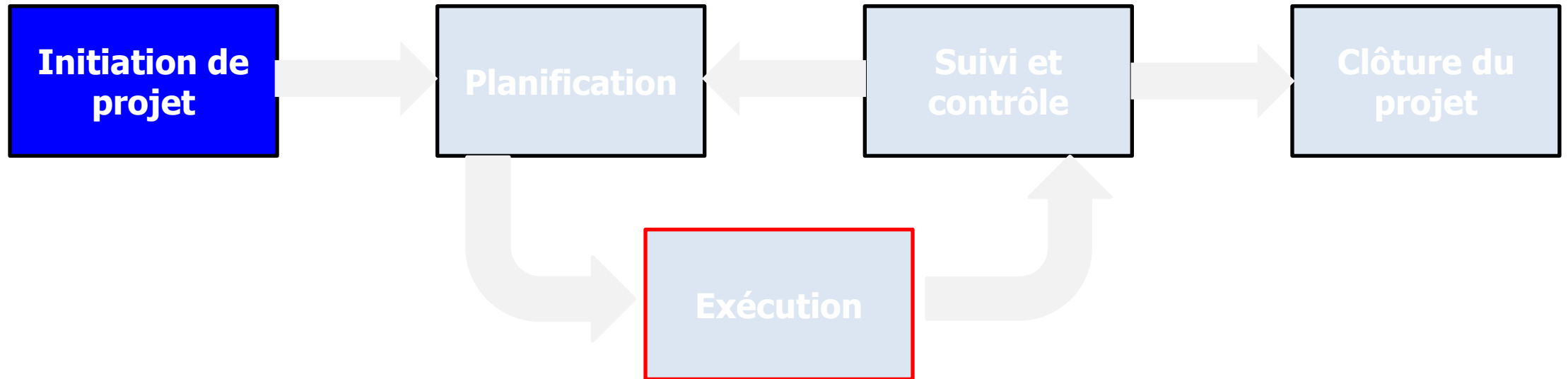
Feuille de route du cours



Introduction

Initiation de projet

1 - Vue d'ensemble



Activités, ressources, outils (IDE, support), normes, contrôles, planification de la gestion du projet

2 - Phase initiale

Motivations derrière un projet

Améliorations ou obligation technique (obsolescence)

Améliorations commerciales et procédurales

La situation idéale est une combinaison des deux aspects

Erreurs à éviter

Démarrer un projet pour suivre une mode

Guider un projet par la technologie (!! Aux standards)

2 - Phase initiale

Identifier la valeur commerciale ajoutée

Etudier la faisabilité

technique

économique

organisationnelle

3 - Environnement technique

1. Choix des outils
2. Mise en place des outils
3. Mise à jour de outils

3 - Environnement technique : Gitlab



GitLab

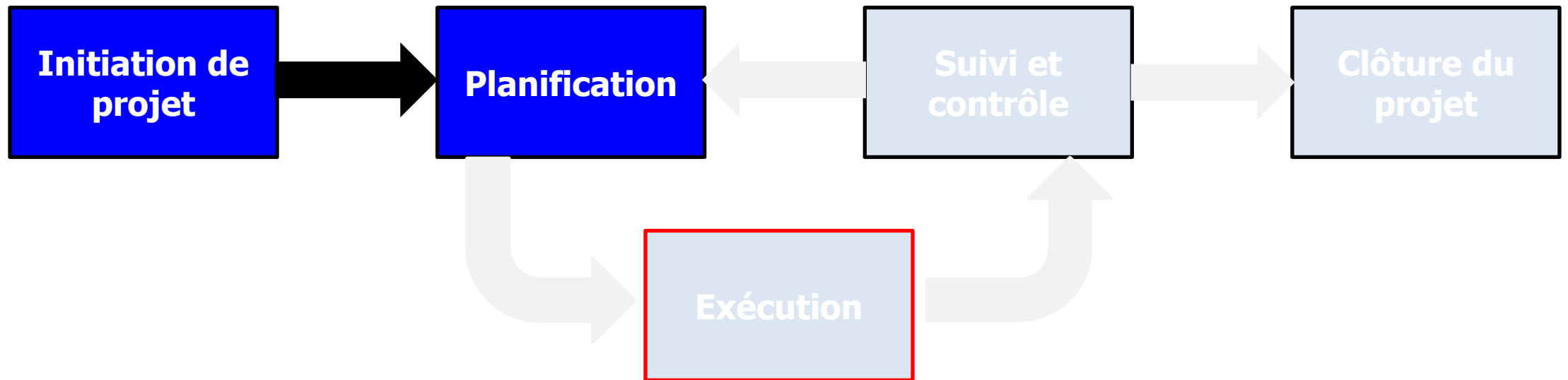
3 - Environnement technique : Gitlab

1. Gestion de projet à distance et collaborative
2. Documentation (Wiki)
3. Code versioning (Git)
4. Suivi de demandes (Issues : bugs, tâches, features...)
5. Planification & suivi (Agile : scrum, Kanban...)
6. ...

Introduction

Planification

1 - Vue d'ensemble



2 - Découpage Top-Down

WBS (Work Breakdown Structure)

Liste hiérarchisée de tâches

4 principes

Règle des 100%

Pas de recouvrement (tâches indépendantes)

Décomposition par résultats, pas par action
(livrables, sous-fonctions, constituants)

Implication des participants*

2 - Bonnes pratiques

La **gestion de projet** est la première tâche (WP0)

Une tâche pour les **spécifications des besoins**

Au moins une tâche pour la **conception**

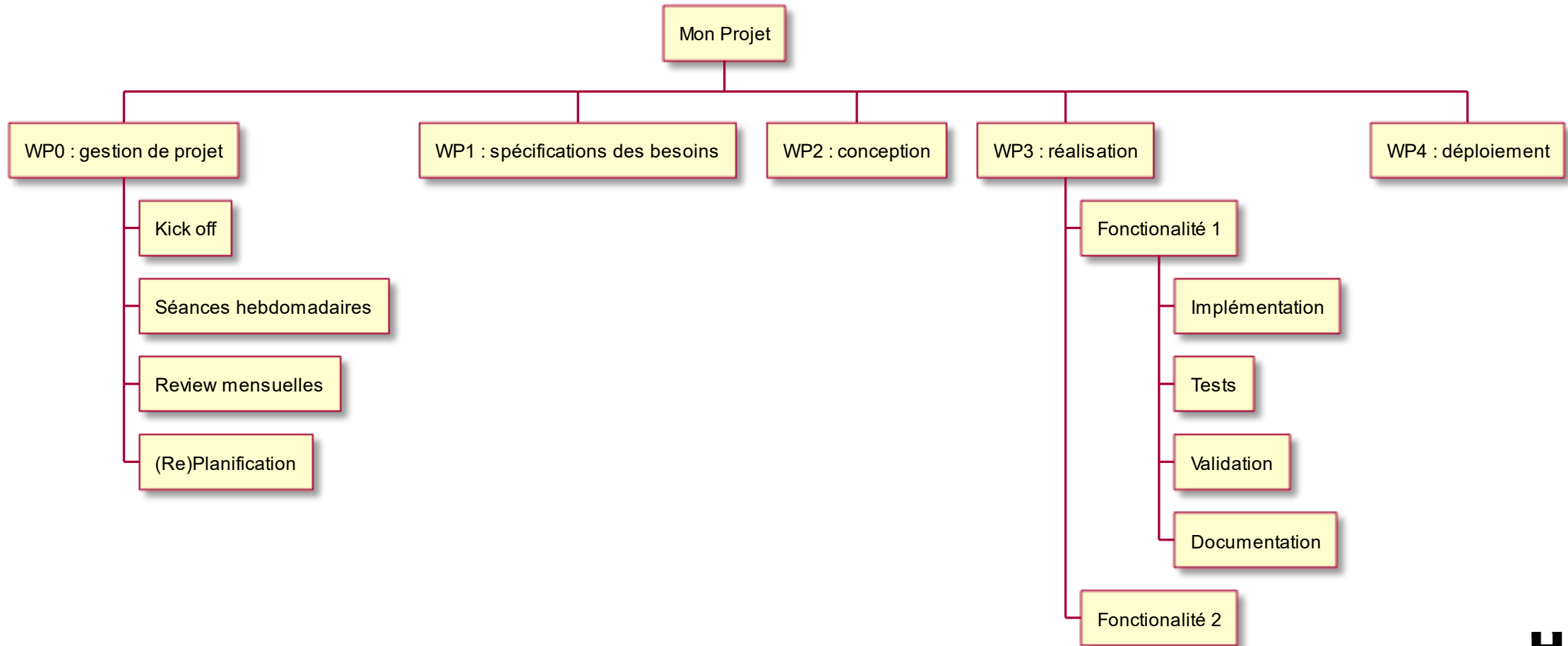
Au moins une tâche pour le **déploiement**

Au moins une tâche pour la **documentation**

Plusieurs tâches pour la réalisation (hardware, software, etc.)

**Discuter de l'organigramme
avec toute l'équipe!**

2 – Exemple de découpage WBS



2 - Jusqu'où découper ?

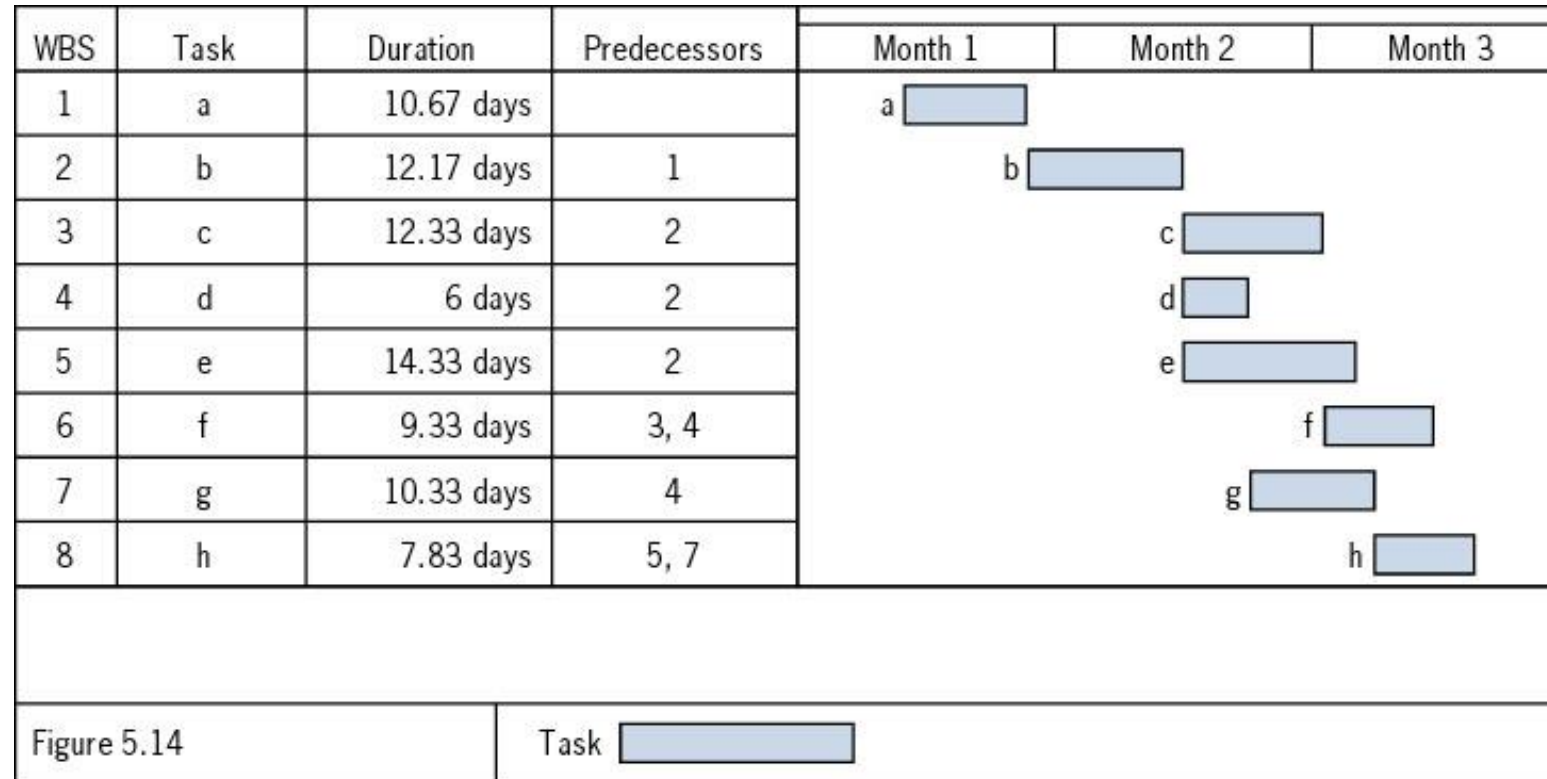
Si on ne peut pas estimer la **charge**

Si on n'a pas la vue d'ensemble des tâches et des étapes à réaliser => **découpage global / général**

Diviser pour régner

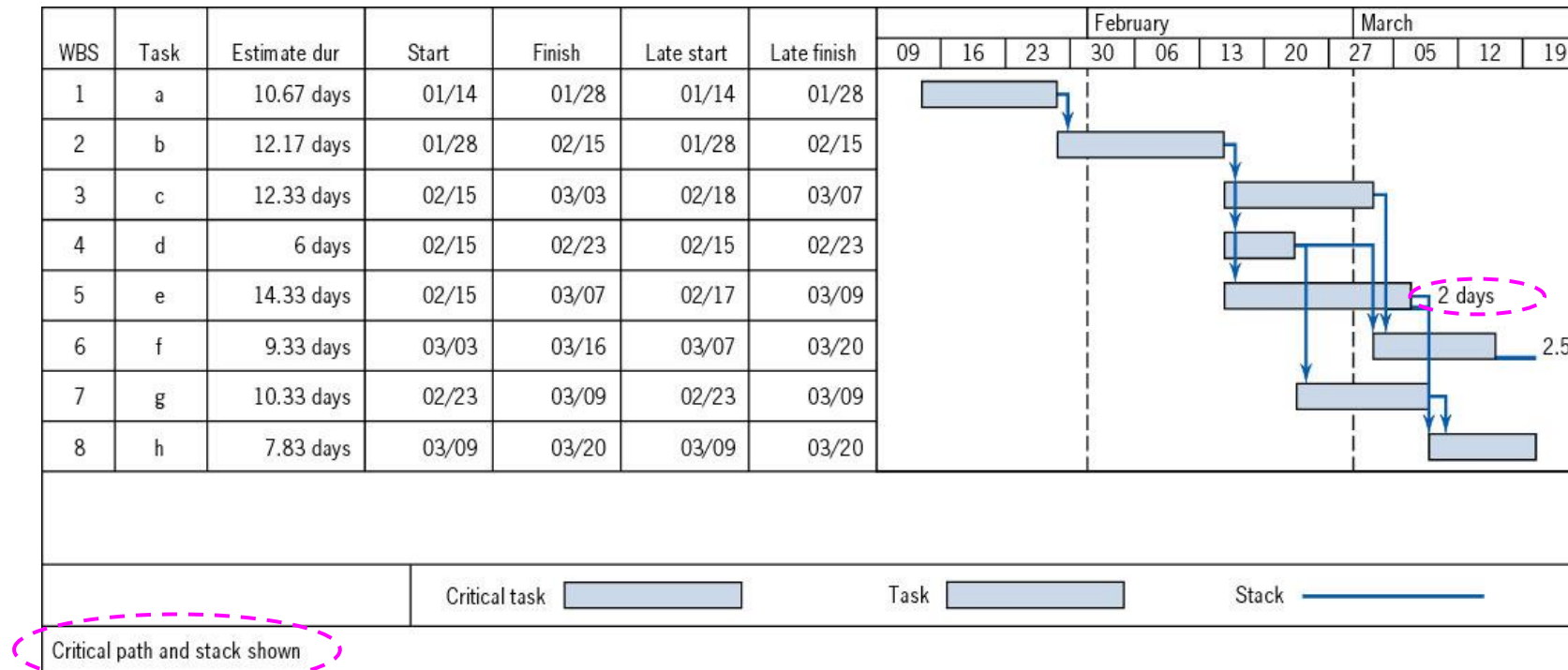
3 - Le diagramme de Gantt

Projet simple



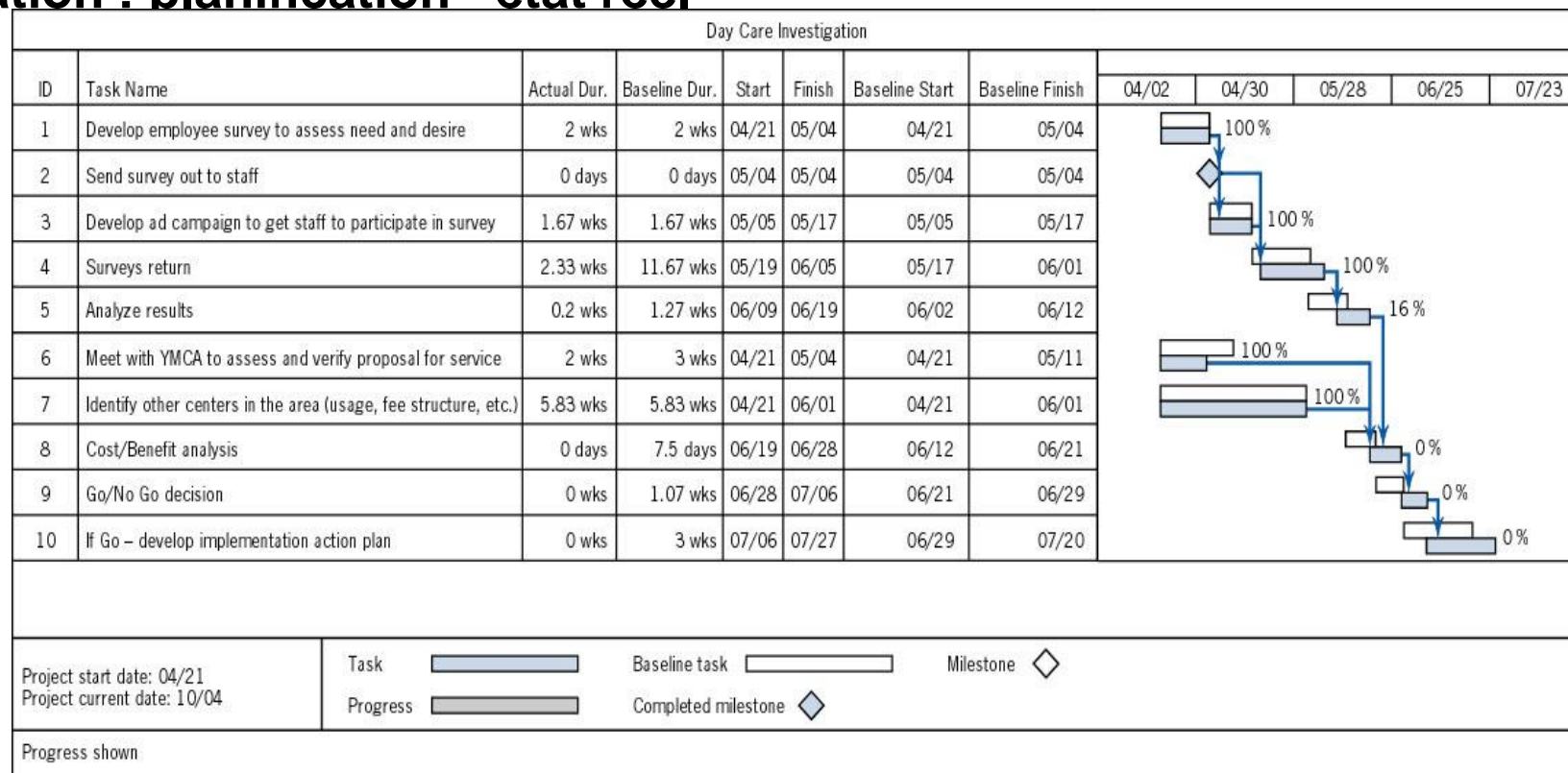
3 - Le diagramme de Gantt

Projet simple avec chemin critique, connexions



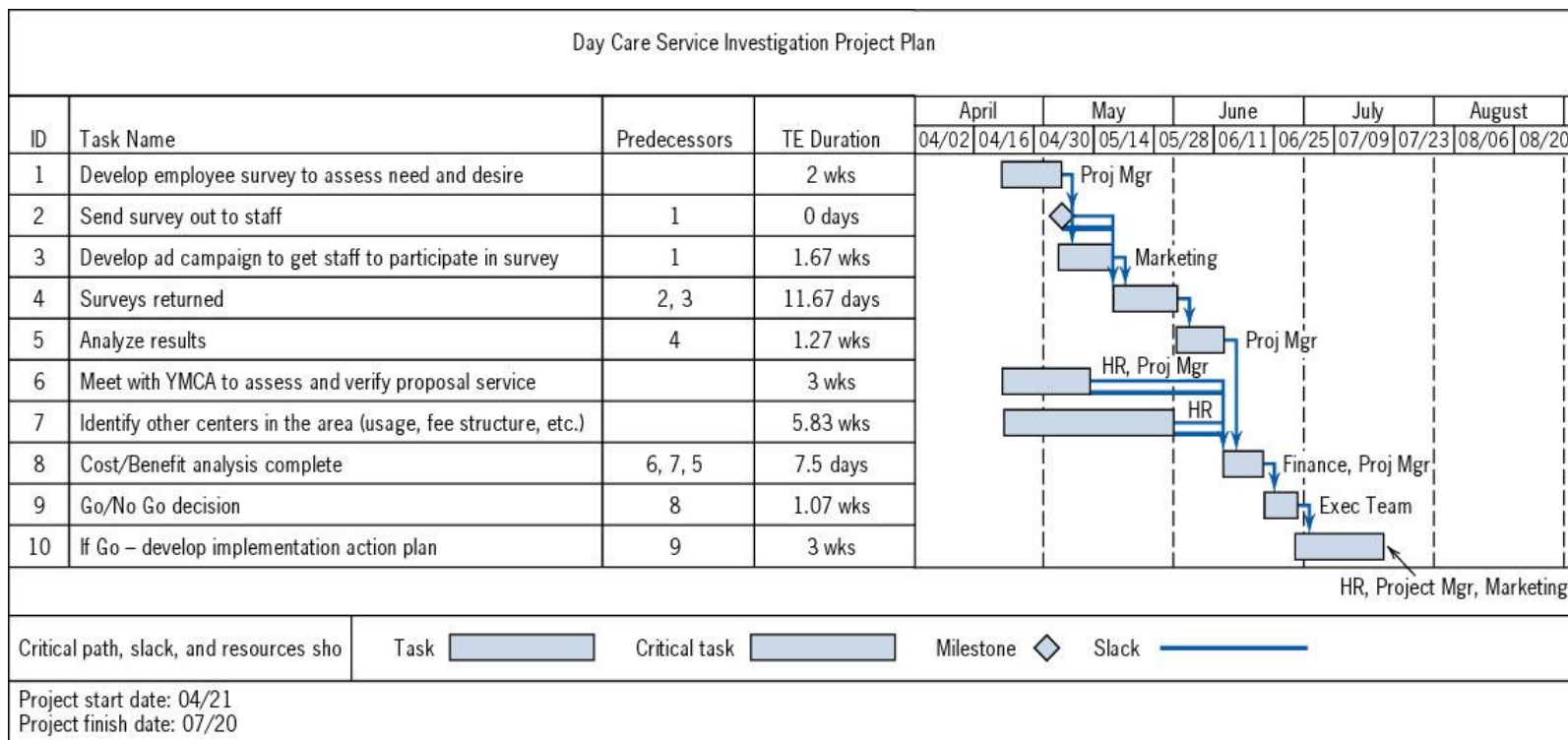
3 - Le diagramme de Gantt

Confrontation : planification - état réel



3 - Le diagramme de Gantt

Durée estimée, chemin critique, jalons (*milestones*) et ressources



4 - Les différents types de risques

Risques financiers	investissements, financement, <i>cash flow</i>
Risques légaux	chgts de lois, IP, protection juridique, garantie,
Risques physiques	catastrophes, décès, météo
Risques intangibles	RH: relationnels, compétences, vie privée
Risques techniques	infrastructure, logiciels, obsolescence, <i>time to market</i>
Risques de sécurité	attaques informatiques, fiabilité, confidentialité

4 - Gestion des risques

Définition

Risque = Problème potentiel

Caractéristiques

Probabilité d'occurrence (P)

Impact (I)

Criticité (C)

Impact	H			
	M			
	L			
		L	M	H
Probability				

The risk matrix is a 4x4 grid. The vertical axis is labeled 'Impact' with levels H, M, L. The horizontal axis is labeled 'Probability' with levels L, M, H. The cells are color-coded: (H,H) is red, (H,M) is red, (H,L) is yellow, (M,H) is red, (M,M) is yellow, (M,L) is green, (L,H) is yellow, (L,M) is green, and (L,L) is green.

$$C = P \cdot I$$

Il faut bien connaître les facteurs de risque qui mènent à ces échecs

7 Suivi de contrôle

Relevé périodique du statut du projet selon les axes suivants:

1. Taux de complétion des tâches / paquets de travail
2. Respects de délais
3. Budget
4. Evolution des risques
5. Problèmes ouverts (*issues*)*
6. Disponibilité de ressources

Communication des rapports aux acteurs concernés

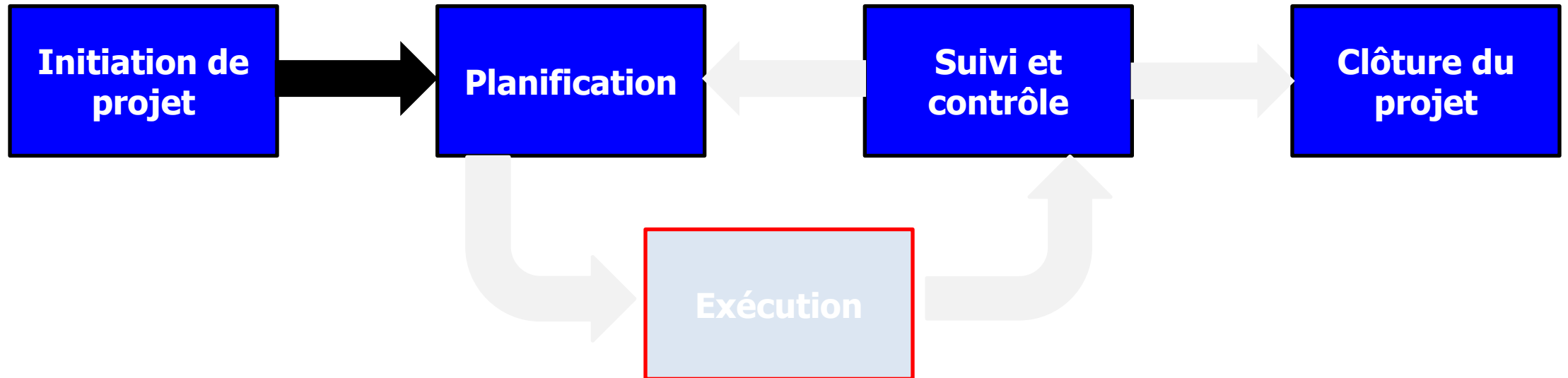
Exemple de rapport de suivi de projet

Overview	Project Description	Pre-study/prototype of overall JAP AR5.0 project. The prototype will deliver a proof of concept to establish the ability of massive transaction requirements in the Java environment.				Major Achievements	• JAP AR5.0 requirements • Proof of Concept • Roadmap for next phase			
Timelines & Major Milestones	Today									
	2007				2008					
	Q1	Q2	Q3	Q4	Q1	Q2	Q3			
				◇ Kick-Off (CW39)						
				◇ JAP AR5.0 requirements (15/Nov)						
					◇ Prototype completed (31/Jan)					
RAG	Milestones Green	Scope Green	Resource Amber	Dependencies Amber	Issues / Decision Green	Cost Green	Risk Amber			
Overall Status Reason	<ul style="list-style-type: none">- All in plan - first Sounding Board meeting was held- Create WBS to get an view what effort is needed to deliver (activity is in progress)- Technical and application concepts started- Set up prototype environment is in progress									
Changes to Scope	- no changes									
Overall Progress	Metric	Target	No. Complete	% Complete	Target Date					
	n/a									
Cost	Status as of Apr 07	Internal Heads (PT)	External Heads (CHF)	Hardware (CHF)	Software (CHF)					
	Forecast (Budget)	250	100'000	0	0					
	Actuals	0	0	0	0					
	Variance	250	100000	0	0					

Introduction

Suivi & Contrôle Cloture

Vue d'ensemble

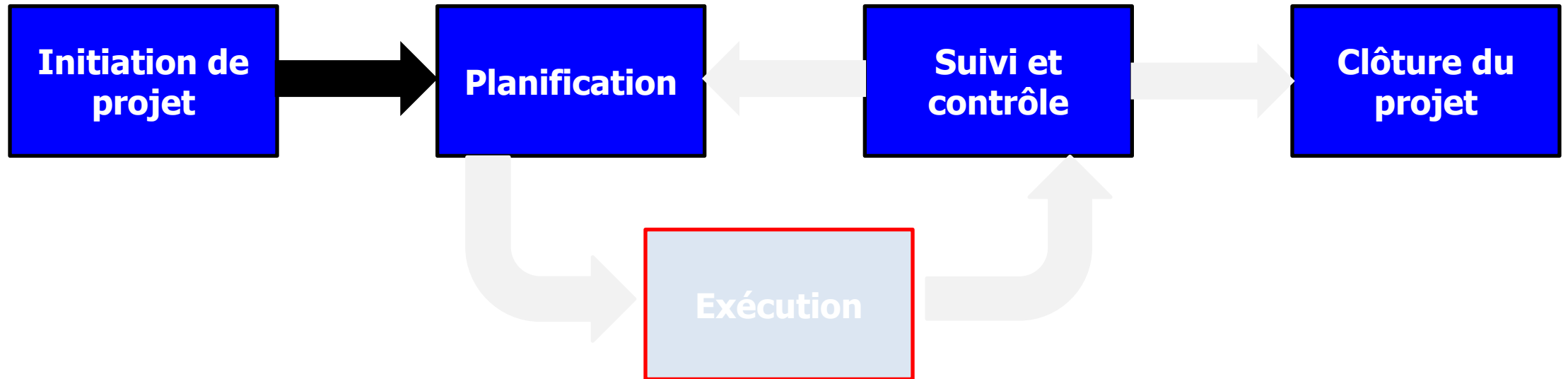


Rapport de clôture – *Lessons learned* -

Lessons Learned Content Detail

Title	Type	What actually happened	Why did it happen	Recommendation
Project set up	+	The project set up : STC, Project Team, Stakeholders, ... was optimally carried out. All concerned parties and stakeholders are involved in a way or another (STC meetings, Requirement gathering, ...). Project approach and decisions are thus syndicated and agreed on before implemented	Project setup deeply analyzed	Involve major concerned parties in the right premum so that they can optimally contribute to the project success
Project approach	+	The project approach allowed to involve RDO (final users) and MPT (support) early in the project lifecycle. Adoption is assured during the project and the handover was optimally prepared	Project approach well designed	Involve final users early in the project lifecycle and manage adoption as integral part (stream) of the project scope.
Project team	+	Project team members showed very complementary skills (Platform engineering, Requirement engineering; system design, ...). Inter-personal relationships worked perfectly	Project skills needs are well understood Good matching between project needs and team	Identify the skills needed by the project Select profiles responding to those skills Team "chemistry" is an important aspect to consider

Récapitulatif



Les phases Planification, exécution et suivi sont effectuées en cycle autant de fois que nécessaire durant le projet.

Exercice : Demande de projet

Écrire une demande de projet
pour votre projet P2