

# Gestion de projets informatiques

---



LICENCE 3 INFORMATIQUE

2022/2023

**Marie NDIAYE DIOP**

# Introduction

---

# Plan

---

Le projet

La gestion de projet

Les acteurs

Les mythes

Echecs/succès

Les conditions de succès

# Le projet

---

# Le projet : définition

---

**Larousse** (<https://www.larousse.fr/dictionnaires/francais/projet/64232>)

- « Étude de conception de quelque chose, en vue de sa fabrication. »

**Ph. Collet**

(<http://deptinfo.unice.fr/twiki/pub/Linfo/ProjetInfo201314/ProjetInfo-Version-1314.pdf>)

- Un effort temporaire qui est progressivement planifié, contrôlé et exécuté par des personnes travaillant avec des contraintes de ressources pour créer un produit, service ou résultat unique.

**ISO10006** (<https://www.iso.org/obp/ui/#iso:std:iso:10006:ed-3:v1:fr>)

- Un projet est un processus unique, qui consiste en un ensemble d'**activités** coordonnées et maîtrisées comportant des dates de début et de fin, entrepris dans le but d'atteindre un **objectif** défini avec des **moyens** adaptés et dans un **délai** donné.

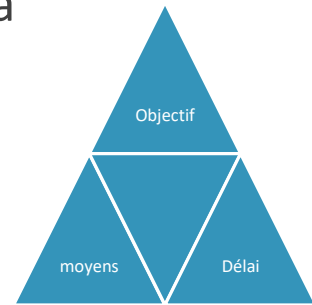
# Le triangle du projet

---

Un projet est parfois représenté par un triangle pour exprimer la solidarité entre les sommets :

Un projet est réussi si :

- L'objectif (satisfaction du client) est atteint;
- Les délais sont respectés;
- Les coûts (moyens) sont respectés.



Si l'un des sommets évolue et que l'on veut conserver le même triangle, il faut agir sur l'un ou les deux autres sommets.

Exemple:

- La variation de l'objectif du projet aura des conséquences soit sur le délai, soit sur les moyens ou les deux.
- La modification des moyens entraîne la modification du délai ou de l'objectif.

# Types de projets

---

**Ouvrage** : résultat unique, un pont, un immeuble, un film, une pièce de théâtre, un logiciel, etc.

**Produit** : mise au point d'une gamme de produit nouveau modèle de voiture, nouvelle création de haute couture, nouvel aliment, etc.

**Opération** : fusion de deux entreprises, automatisation des processus, formation des personnels

**Événement** : Coupe du Sénégal de Football, Foire de Dakar, Forum du numérique

# Le projet : 5 aspects

---

**Fonctionnel** : réponse à un besoin

**Technique** : respect des spécifications et des contraintes

**Organisationnel** : respect d'un mode de fonctionnement (rôles, culture, fonctions, résistance au changement)

**Délais** : respect des échéances (planning)

**Coûts** : respect du budget



# Projet informatique

---

**Sokeo** (<https://sokeo.fr/quest-ce-quun-projet-de-developpement-informatique/>)

- Un projet informatique est un projet de développement d'un nouveau logiciel ou de l'installation d'une solution système, ainsi qu'un ERP (progiciel de gestion intégrée) ou un CRM (logiciel de gestion de relation client).
- Ce type de projet nécessite plusieurs étapes à suivre : Recueil besoins, Cahier des charges, Spécification, Création, Conception, Développement, Livraison, Tests, etc.
- Chaque étape doit aboutir à la mise à disposition de livrables.

# La gestion de projet

---

# C'est quoi la gestion de projet?

---

La **gestion de projet** est une démarche qui vise à organiser et à surveiller le déroulement d'un projet, par la mise en œuvre de connaissances, de compétences, d'outils et de techniques appliqués au projet, afin d'atteindre les objectifs visés.

Aux trois aspects d'un projet (Objectif, Moyens, Délai) correspondent trois types de gestion :

- **Gestion de la production** – Suivre et diriger l'avancement vers l'atteinte des objectifs tout au long du projet
- **Gestion des ressources** – Elle inclut la gestion des moyens humains, la gestion du matériel et la gestion du budget (coûts d'utilisation des ressources)
- **Gestion du temps** – Mettre en place des calendriers et veiller à ce que les échéances soient respectés

# Gestion de projet : Pourquoi?

---

On ne peut même pas se faire confiance ! Alors imaginer avec une équipe !

On ne peut pas ne pas faire de bug (moi aussi j'ai cru que je ne faisait pas de bug!)

Très difficile

- d'évaluer le temps exact (problèmes conceptuels, bugs...)
- de prévoir les problèmes (Incompatibilité entre 2 logiciels)
- d'anticiper les difficultés (La base de données client est mal faite)
- d'imaginer les aléas (Le serveur vient de se planter)

# Enjeux et motivations

---

**La réussite des projets relatifs aux Systèmes d'Information (SI) est une nécessité.**

- Les dépenses informatiques représentent près de 30% du chiffre d'affaire dans certains secteurs d'activité (banques en particulier). Elles constituent souvent la deuxième sources de dépense après les salaires.
- Les entreprises investissent dans l'informatique car le bon fonctionnement des SI et leur **capacité à évoluer vite** en accompagnant la stratégie de l'entreprise sont vitaux :
  - La plupart des grandes entreprises du tertiaire auraient du mal à survivre si leur SI s'arrêtait de fonctionner pendant 3 jours.
  - Les processus de décision voire de production sont supportés par des SI : Il est presque impossible de les faire évoluer si les SI ne suivent pas.
  - La vitesse d'évolution des SI est un facteur différenciant fort par rapport à la concurrence.

**Les projets informatiques sont de plus en plus gros faisant intervenir des gens d'horizons différents.**

# Les acteurs

---

# MOA et MOE

---

**MOA** – Le maître d'ouvrage (L'**ouvrage** est le résultat concret d'un projet)

- La personne physique ou morale ( direction etc. qui sera le propriétaire de l'ouvrage). Il fixe les objectifs, l'enveloppe budgétaire et les délais souhaités pour le projet.

**MOE** – Le maître d'oeuvre (L'**oeuvre** est le processus de réalisation de l'ouvrage)

- La personne physique ou morale ( direction, etc.) qui réalise l'ouvrage pour le compte du maître d'ouvrage et qui assure la responsabilité globale de la qualité technique, du délai et du coût.
- Lorsque le produit est complexe, il peut être nécessaire de faire appel à plusieurs fournisseurs. Dans ce cas, le MOE assure la coordination des fournisseurs en contrôlant la qualité technique et en assurant le respect des délais et du coût fixés par le MOA.

# Utilisateur

---

son rôle est :

- d'exprimer les besoins et les contraintes à au début du projet
- de valider le produit final (s'assurer qu'il est conforme à au cahier des charges).
- Dans certain types de projets, l'utilisateur peut intervenir dans les étapes intermédiaires :
  - Aider à la précision des besoins ou des contraintes;
  - Effectuer des validations partielles;
  - Etc.



# Chef de projet et équipe projet

---

## **Chef de projet**

- La personne physique chargée par le maître d'œuvre d'assurer la maîtrise du projet, c'est à dire de veiller à sa bonne réalisation dans le respect des objectifs de technique, de coût et de délai
- Il est chargé entre autres
  - de la définition, de la planification et du pilotage du projet
  - des négociations internes et externes au projet (avec les partenaires)
  - de l'animation des équipes

## **Equipe projet**

- L'ensemble des personnes placées sous l'autorité directe du chef de projet:
  - Concepteur: Responsable de la conception du futur système.
  - Développeur: Responsable du codage des programmes ou de réalisation de prototypes.

# Sous-traitant et expert externe

---

## **Sous traitant**

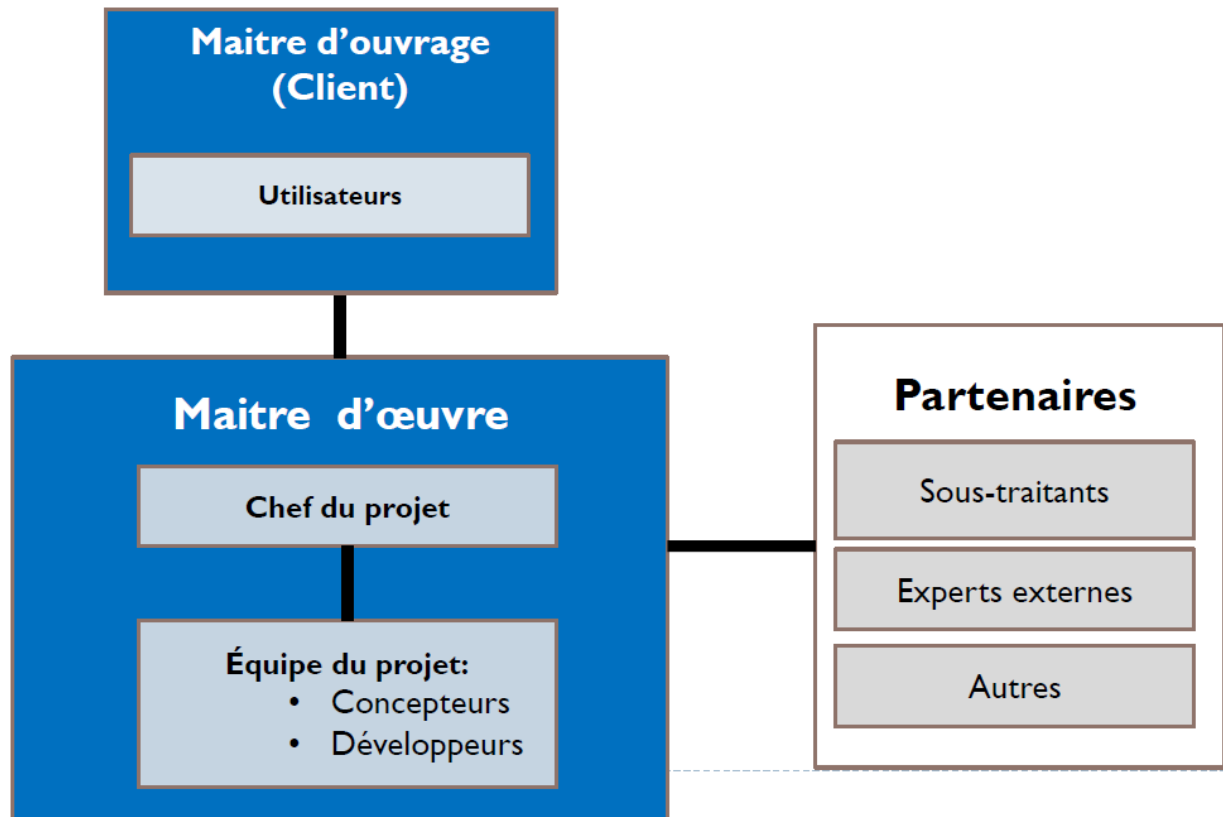
- Un sous traitant est un acteur externe chargé de la réalisation d'une partie du projet.

## **Expert externe**

- Un expert est quelqu'un qui a une compétence dans un domaine. Il est sollicité par le maitre d'oeuvre pour traiter un problème précis (estimation de charges, planification, etc.).

# L'ensemble des acteurs

---



Source : [http://elearning.centre-univ-mila.dz/pluginfile.php/79762/mod\\_resource/content/1/GPI-Chapitre%20I.pdf](http://elearning.centre-univ-mila.dz/pluginfile.php/79762/mod_resource/content/1/GPI-Chapitre%20I.pdf)

# Les mythes

---

# Mythes du chef de projet

Les outils actuels sont la solution

- Un nul avec un outil est toujours un nul

Si on est en retard, on ajoutera du personnel



# Mythes du client

---

Une idée générale des objectifs est suffisante pour commencer le codage – on ajoutera les détails plus tard.

- Une forte communication entre clients et développeurs est toujours nécessaire.

Les changements peuvent être facilement répercutés parce que le logiciel est flexible

- Les changements ne peuvent être évités, c'est la vie...
- Les changements tardifs coûtent très chers

# Mythes du développeur

---

Une fois que le programme est écrit et qu'il tourne, le travail est terminé.

- Certainement pas!

Jusqu'à ce que le programme tourne, il n'y a aucun moyen d'évaluer sa qualité.

- Inspections & revues

La seule chose à livrer pour un projet réussi est un programme qui marche.

- Documentation, Tutorial ...

# Echecs/succès

---

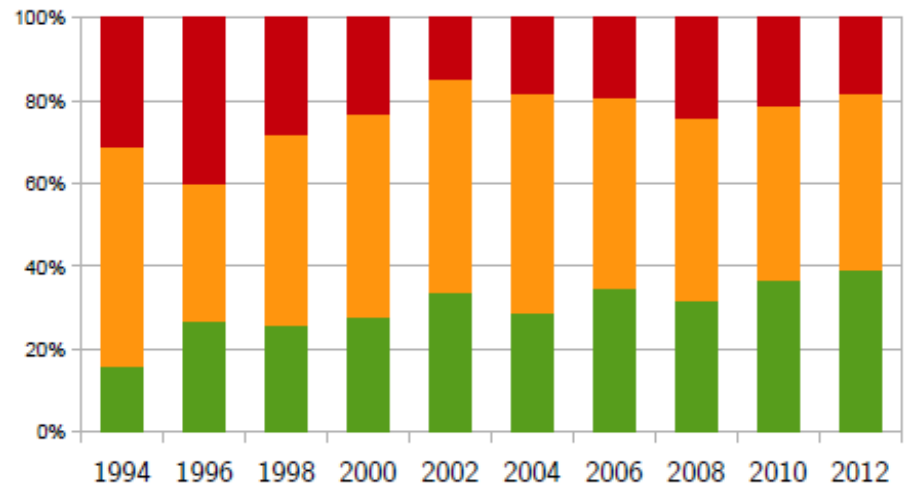


# Echec/Succès : Le constat

## Minimiser les risques d'échec

Enquête sur des milliers de projets, de toutes tailles et de tous secteurs

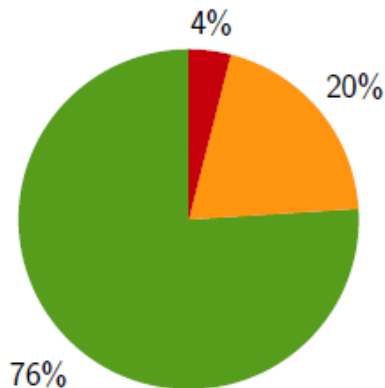
- **Projets réussis** : achevés dans les délais et pour le budget impartis, avec toutes les fonctionnalités demandées
- **Projets mitigés** : achevés et opérationnels, mais livrés hors délais, hors budget ou sans toutes les fonctionnalités demandées
- **Projets ratés** : abandonnés avant la fin ou livrés mais jamais utilisés



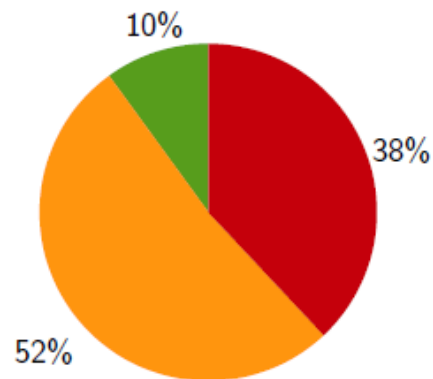
Standish group, *Chaos Manifesto 2013 - Think Big, Act Small*, 2013

# Echec/Succès : Le constat

Petits projets/ grands projets

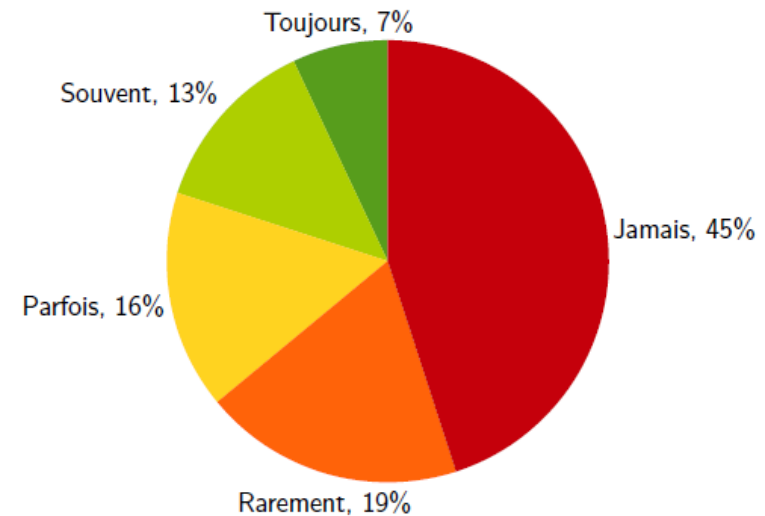


Petits projets  
budget  $\leq$  \$1 million



Grands projets  
budget  $\geq$  \$10 millions

Utilisation des fonctionnalités implantées



Standish group, *Chaos Manifesto 2002*, 2012

Standish group, *Chaos Manifesto 2013 - Think Big, Act Small*, 2013

# Quelques exemples d'échec

---

**Surcoût** : L'informatisation du paiement des indemnités de sécurité sociale en Grande-Bretagne: budget estimé à 7 milliards de francs, révisé à 12, puis 16, puis 18 milliards. Le projet a finalement coûté 26 Milliards ! Soit une augmentation de 300%.


**Retard** : London Ambulance Service avait à l'origine prévu un délai d'un an pour informatiser la gestion de son parc d'ambulances. Quatre ans après, le système ne fonctionnait toujours pas.

**Surestimation des bénéfices** : Le ministère américain des Anciens Combattants pensait que son nouveau système réduirait le temps de traitement des dossiers de 4 mois à 6 semaines. 2 ans après la mise en place du système, il fallait 8 mois pour traiter un dossier !

**Manque de fiabilité** : Tiptree, un distributeur de livres britannique a développé un système de gestion de stocks. Quelques semaines après sa mise en service, le système avait perdu la trace de la majorité des 25 millions d'ouvrages en stock, à la suite d'une accumulation de petites erreurs. Il a fallu 2 ans pour retrouver une qualité de service équivalente.

# Les causes d'échec

---

1. Manque de contribution des utilisateurs 12,8%
2. Exigences et spécifications incomplètes 12,3%
3. Modification des exigences et des spécifications 11,8%
4. Manque de soutien exécutif 7,5%
5. Incompétence technologique 7,0%
6. Manque de ressources 6,4%
7. Attentes irréalistes 5,9%
8. Objectifs non clairs 5,3%
9. Cadrage de temps irréalistes 4,3%
10. Nouvelle technologie 3,7% 
11. Autres 23,0%

- **La technologie informatique est récente** dans l'histoire de l'humanité : moins de 40 ans d'utilisation intensive (à comparer au bâtiment et travaux publics : plusieurs siècles).
- **Le cycle de renouvellement trop rapide des technologies =>** difficulté de capitaliser sur l'expérience et maîtrise plus difficile.

# Les mêmes erreurs sont généralement mises en cause

---

## Une étude de Computer Weekly

- Un à priori selon lequel l'informatisation est toujours une bonne chose
- Une ambition excessive pour le projet
- La volonté de développer un système intégré
- Une préférence pour les systèmes développés sur mesure au détriment de systèmes qui ont déjà fait leurs preuves
- La fierté d'informaticiens qui ne veulent pas montrer leur limites
- Une propension à ignorer les problèmes
- La dissimulation des problèmes ou de leur gravité par l'équipe de projet
- Le manque de courage du PDG qui ne prend pas ses responsabilités à l'égard du projet parce qu'il s'estime ignorant en informatique
- L'abdication de la responsabilité du projet au profit du fournisseur
- Des délais imposés trop courts

# Les conditions de succès

---

# Les coûts

---

	Conception	Réalisation	Vie de l'application
<b>Coût total =</b>	1	3	9

Coût de Réalisation = 3 x Coût Conception; Coût Vie du produit = 3 x Coût Réalisation.

<b>Coût impasse =</b>	1	10	100
-----------------------	---	----	-----

1 jour « économisé » en conception coûte 10 j en réalisation et 100 jours en production.

# Tenir compte du Coût total : projet + production

---

La réussite se mesure sur l'ensemble projet/après-projet car la vie des produits représente 70% de leur coût.

- La réussite d'un projet est donc bien plus que mener à bien la conception et la réalisation. La vie du produit doit faire partie intégrante des préoccupations.

C'est le cycle complet qu'il est important de rendre efficace pour l'entreprise et non seulement la phase Projet.

Cependant, les coûts en phase de Vie du produit sont avant tout dépendants du **niveau de qualité** atteint par le Projet.



# Exemples

---

Un projet qui se terminerait « bien » en entraînant des surcoûts énormes par une utilisabilité médiocre.

On fait des économies de bouts de chandelle pendant le projet (pour tenir les coûts et/ou les délais, on rogne sur l'étude et/ou l'implémentation de fonctionnalités structurantes) et ces fonctionnalités indispensables sont à introduire dans des versions suivantes.

# Privilégier l'amont

---

1 jour faussement économisé en conception risque de coûter 10 jours en réalisation et 100 jours pendant la vie du produit.

Les rapports 1/10/100 ne sont que des ordres de grandeur et ne doivent donc pas être pris à la lettre mais servir plutôt à apprécier où et quand doivent porter les efforts : il est plus facile de modifier les plans d'un immeuble en prévision d'évolutions futures que de rajouter un étage quand l'immeuble est fini.

- **Les impasses conceptuelles coûtent très cher** : Ce sont les choix effectués pendant le projet, et notamment la qualité des phases AMONT (conception), qui sont primordiaux.
- **L'évolutivité, la souplesse et l'adaptabilité du produit sont un résultat direct des phases en amont.**

Ce sont donc ces phases qui sont source d'économies à moyen/long terme.

# Exemples

---

Cahier des charges de la sécurité du tunnel sous la manche évoluant pendant la réalisation.

carte crédit JAZZ : le compteur de points gagnés a été associé à la carte au lieu du compte client.

- Impossible de cumuler les points si le client a plusieurs prestations.

# Employer une méthode renforçant les phases Amont

---

La méthode est davantage un recueil de bonnes pratiques qu'une méthodologie.

Attention : il convient de s'imprégner de l'esprit plutôt que l'appliquer à la lettre (privilégier le fond à la forme).

Il ne faut jamais oublier que l'objectif d'un projet est sa réussite (la fin) et non l'application d'une quelconque méthode (le moyen).

Mais il ne faut pas oublier non plus que la méthode est le fruit des expériences passées, qu'il faut garder présentes à l'esprit.

# Ne pas appliquer la méthode aveuglement

---

Tenir compte de la typologie du projet en le classant sur un certain nombre d'axes :

- Complexité du sujet
- Expérience de l'équipe, diversité/implication des acteurs
- Taille du projet
- Technologies éprouvées
- Stabilité du contenu

# Se méfier d'une ambition excessive

---

## **Ne lancer un projet qu'en dernier recours**

- Envisager les alternatives à un nouveau développement.
- Valider que le projet répond à un besoin précis.

## **Faire simple**

- Préférer les progiciels aux solutions sur mesure.
- Préférer des systèmes modulaires plutôt qu'intégrés.
- S'en tenir aux besoins fondamentaux.
- Etre ferme sur les changements ultérieurs de spécifications.

# Se préparer à des dérapages

---

Ne pas fixer des délais artificiellement serrés

Multiplier par 2 les estimations de coût

Observer d'autres systèmes

Border son contrat avec les fournisseurs

# Assurer une gestion vigilante du projet

---

## Vigilance

- Pilotage rigoureux
- Souci constant de la qualité : soigner la documentation
- Prévoir la possibilité d'arrêter le projet à différents stades
- Cultiver le scepticisme



# Une direction forte

---

La Direction doit prendre ses responsabilités.

Ne pas abandonner la responsabilité du projet au prestataire.

Bien choisir le chef de projet. Il doit avoir les compétences suivantes :

- Une connaissance de l'activité suffisante pour savoir distinguer les besoins cruciaux des demandes superflues.
- Une indépendance et une confiance en lui suffisantes pour rejeter tout changement de spécifications ultérieur qui ne serait pas vital.
- La capacité d'écouter chacun.
- L'honnêteté nécessaire pour reconnaître ses erreurs et le courage de transmettre des mauvaises nouvelles à la Direction.
- Le réalisme et le courage de vouloir faire simple même si cela paraît moins valorisant.

# Le succès

---

Un Projet Réussi est un projet bien géré qui atteint son objectif

- L'**objectif** doit être atteint : application conforme aux attentes
- **Délai** : le projet doit être réalisé dans les délais impartis
- **Budget** : il ne doit pas être dépassé

Juger la Qualité du Produit

- La qualité du produit du projet (l'application) doit être évaluée, comme celle de n'importe quel produit technique, suivant certains critères :
  - Facilement utilisable ?
  - Fiable ?
  - Durable ?
  - Facile d'entretien ?

La **qualité du produit** dépend essentiellement de celle du processus, c'est à dire du **Projet**.

# Références

---

[http://elearning.centre-univ-mila.dz/pluginfile.php/79762/mod\\_resource/content/1/GPI-Chapitre%20I.pdf](http://elearning.centre-univ-mila.dz/pluginfile.php/79762/mod_resource/content/1/GPI-Chapitre%20I.pdf)