

## Le business plan d'un projet informatique

### Etude des coûts prévisionnels et du retour sur investissement escompté

Version	V1.0
Date	Janvier 2022
Auteur	Jean-Pierre ROBERTI

# Le business plan d'un projet informatique

Ce document est publié sous la licence libre **Creative Commons-BY-NC-SA**  
<http://creativecommons.org/licenses/by-nc-sa/2.0/fr/>



**BY : Paternité.** Vous devez citer le nom de l'auteur original.

**NC : Pas d'Utilisation Commerciale.** Vous n'avez pas le droit d'utiliser cette création à des fins lucratives et commerciales.

**SA : Partage des Conditions Initiales à l'Identique.** Si vous modifiez, transformez ou adaptez cette création, vous n'avez le droit de distribuer la création qui en résulte que sous un contrat identique à celui-ci. En outre, à chaque réutilisation ou distribution, vous devez faire apparaître clairement aux autres les conditions contractuelles de mise à disposition de cette création.

Chacune de ces conditions peut être levée si vous obtenez l'autorisation du titulaire des droits.

# Le business plan d'un projet informatique

- Sommaire
  - Quelles sont les Particularités d'un business plan ?
  - Quelles sont les causes d'un écart concernant les coûts ?
  - Comment s'organiser pour gérer ces aléas ?

## Les particularités du business plan d'un projet informatique

- **La difficulté principale réside dans le chiffrage des coûts et gains prévisionnels**, souvent beaucoup plus aléatoires pour un projet informatique que pour la moyenne des autres projets.
- On constate parfois un écart de 1 à 5 entre les coûts initialement prévus et les coûts réels, sans que la compétence technique du responsable de projet soit en cause.
- Ce dernier est souvent passionné au plan technique et n'a pas toujours la prise de recul suffisante pour évaluer l'étendue du développement. C'est sur ce point que les financiers peuvent l'aider.

## Causes d'écart concernant les coûts

On identifie **6 grandes causes** de dérive des coûts d'un projet informatique par rapport à la définition du périmètre :

- 1. La définition des fonctionnalités attendues** : Il est quasi inévitable que le client demande des fonctionnalités supplémentaires en cours de développement. Cette demande constitue évidemment une cause de dérive des coûts par rapport au cahier des charges initial ;
- 2. Les besoins techniques, hors fonctionnalités** n'ont pas été bien précisés ou ont évolué dans le temps : performance attendue, volumétrie, règles de confidentialité,...
- 3. La nécessité de réaliser des prestations supplémentaires non prévues** au cahier des charges préalables au développement informatique. Exemple : Si les processus métier que l'on veut informatiser n'ont pas déjà été formalisés, il faudra préalablement les formaliser.

## Causes d'écart concernant les coûts

### 4. **Des prestations liées à la mise en place de l'application.**

Les prestations annexes à l'installation ont-elles toutes été prévues, par exemple, la reprise des données, un double run (le double run consiste à faire tourner en parallèle l'ancienne et la nouvelle application en phase de lancement, par sécurité).

### 5. **L'étendue des obligations liées à l'après-vente : la garantie, le support hotline ont-ils été prévus ?**

**6. Développements liés au périmètre géographique.** Dans les groupes, les applications ont aujourd'hui un champ d'utilisation mondial. Lorsque des tranches horaires d'utilisation de l'application ont été définies, l'application doit tenir compte des différents fuseaux horaires sur lesquels se situent les utilisateurs.

## Causes d'écart concernant les gains escomptés

Le niveau d'aléa sur les gains futurs ( = rentabilité du projet, = retour sur investissement ) dépendra du type de projet.

Ces gains consistent en des revenus supplémentaires (chiffre d'affaires) ou des économies (réductions de charges).

**On distingue habituellement 4 types de projets informatiques :**

**Premièrement, le projet vise à améliorer la qualité de service et réduire les risques.** Exemple : L'installation d'une plate-forme haute disponibilité diminue le risque de panne. Les gains sont assez difficilement chiffrables car le bénéfice est la réduction d'un risque dont il n'est pas certain qu'il se réalise.

**Deuxièmement, le projet vise à réduire les coûts du service informatique.** Exemple : La direction informatique décide d'utiliser des logiciels « open source », libres de tout droit de licences. Le projet comporte toutefois des coûts d'installation et de formation. L'estimation des coûts et gains est à priori relativement aisée.

## Causes d'écart concernant les gains escomptés

**Troisièmement, le développement contribue à la productivité des utilisateurs.**

Les gains se situent cette fois au niveau des services utilisateurs et non du service informatique. Exemple : développement d'une nouvelle application permettant une automatisation des tâches, la suppression de double saisies,....Ici encore, l'estimation des coûts et économies est à priori peu aléatoire.

**Enfin, le développement porte sur une innovation permettant de générer du chiffre d'affaires supplémentaire** en proposant un service nouveau ou en accroissant son avantage concurrentiel.

Exemples : application favorisant des achats supplémentaires sur un site marchand.

La détermination des revenus supplémentaires repose sur une estimation qu'il n'est pas trop difficile de réaliser.



## Comment s'organiser pour gérer ces aléas ?

**Pour chaque jalon, le chef de projet mesure l'écart entre les coûts réels et les coûts prévus.**

Quand les coûts réels commencent à diverger de façon importante des coûts prévus, il s'efforce de les faire converger de nouveau. Il utilise pour cela différents moyens :

- **Il intègre des « contingences » dans son offre initiale** sur différents éléments : temps de développement, achats de prestations, niveau de qualité demandé,....
- **Il va également proposer de décaler la livraison des fonctionnalités qui ont moins de valeur pour le client.**
- **En cas de dépassement des coûts ou délais, le développement de ces fonctionnalités pourra être reporté sur une version ultérieure.**

**A la fin du projet, il détecte les sources d'écart pour en tenir compte pour les projets futurs.**