**UC D51.2 Conduite de Projets**

**Dossier 1 :**

**Question 1 :**

1. Une mauvaise communication

Une mauvaise communication est la principale raison pour laquelle un projet n'atteint pas le résultat souhaité ou, tout simplement, échoue. Il peut y avoir plusieurs causes :

* Les barrières linguistiques
* Des traits de caractère qui s'opposent
* Des facteurs liés à l'environnement professionnel
* Des canaux défaillants

Les conséquences d'une mauvaise communication sont également multiples :

* Des attentes différentes (et donc une déception)
* Des égos froissés
* Du stress
* Un manque de vision globale
* Des requêtes de dernière minute
* Un travail bâclé

2. Une mauvaise planification

La planification est parfois considérée comme une activité secondaire, une chose qui doit être faite mais qui n'a aucune valeur ajoutée. Mais si vous ne parvenez pas à définir précisément le périmètre, les responsabilités et le calendrier de votre projet, vous aurez de sérieux problèmes par la suite. Sans un bon plan d'actions, vous risquez :

* Un manque de soutien de la part de l'entreprise
* Un manque de ressources et de temps
* Le non-respect des délais
* Des membres de l'équipe ou des clients mécontents

3. Passer à côté de certaines informations

Combien de budget vous reste-t-il ? Qui est responsable de quoi ? Êtes-vous toujours dans les délais pour pouvoir tenir vos échéances ?

Chacun de nous appréhende les silences embarrassants lors des réunions sur l'évolution du projet ou, pire encore, de parler du mauvais projet au mauvais client. Au minimum, le fait de ne pas en connaître les détails n'inspire pas confiance. Au pire, tout votre projet s'écroule.

4. Utilisation des mauvais outils

Les chefs de projet utilisent souvent plusieurs outils pour faire le suivi des tâches, des membres de l'équipe, des livrables et de l'état d'avancement. Cependant, dans de nombreux cas :

* Votre logiciel n'est pas assez souple pour vous accompagner dans la gestion de votre projet
* Sa complexité vous ralentit

Le résultat peut vous sembler sans queue ni tête.

5. Manque de surveillance et de contrôle

Supposons que vous avez commencé à travailler sur un projet et créé un calendrier solide, mais vous ne mettez jamais votre planning à jour. Il ne vous faudra pas longtemps pour en perdre la vision d'ensemble car :

* Les échéances ont changé
* Les activités ont été interverties
* Les ressources ont été réaffectées

Pour que tout se déroule comme prévu, vous devez effectuer des contrôles réguliers et faire les changements qui s'imposent. Vous devez donc également faire des points réguliers avec votre équipe pour voir comment éviter que le projet ne prenne du retard.

6. Manque de gestion des risques

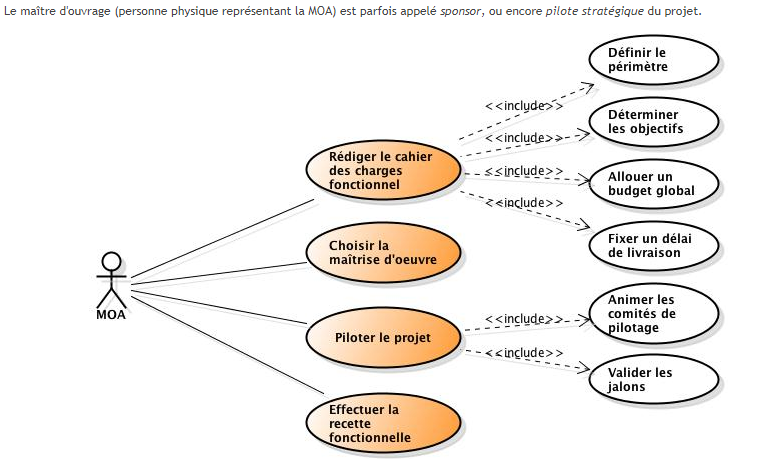
Chaque projet étant unique, l'imprévisibilité devrait toujours être considérée comme un facteur inhérent. La qualification et la quantification de ce facteur, l'anticipation proactive de ce qui pourrait mal tourner et votre réaction si cela arrivait s'appelle la « gestion des risques ». Et oui, cela fait également partie du travail d'un chef de projet.

7. Modifications du périmètre (ou dérive des objectifs)

Scénario classique : une partie prenante du projet demande un « petit changement » ou un « ajout mineur » au périmètre. Si petit, en fait, que vous ne pensez pas que le budget ou les délais en seront affectés. Pas vrai ? Faux !

Chaque projet est un équilibre fragile entre les éléments de la « triple contrainte » : objectif, délai et coût. Sachez que la modification de l'un de ces facteurs affecte aussi automatiquement l'un des deux autres (ou les deux) : si l'objectif change, les délais sont reportés ou le coût de votre projet augmente.

**Question 2 :**

****

L’**Assistance à Maîtrise d’Ouvrage** – **AMOA** a pour mission d’assister la MOA en mettant en œuvre tout au long de sa mission des moyens et des compétences pour aider ce dernier à atteindre ses objectifs.

L’**AMOA** va assurer la **coordination** avec le **MOA**, Il va participer à la rédaction des spécifications fonctionnelles, valider les aspects fonctionnels, manager les équipes de production, assurer les phases de recette, produire les documents de formation et réaliser la formation du MOA.

Le rôle de l’Assistance à Maîtrise d’Ouvrage n’est pas à négliger lors d’un projet d’acquisition et de déploiement d’un logiciel.

Il doit en effet :

* Décrire le besoin utilisateur en définissant le « pourquoi » du projet et les changements souhaités dans le futur système d’information.
* Définir la solution fonctionnelle en décrivant la solution logicielle envisagée par la rédaction d’un cahier des charges – CDC ou de spécifications – qui répondra aux besoins des utilisateurs.
* Contrôler la qualité en vérifiant que le logiciel développé correspond bien au besoin de départ.
* Piloter le projet en suivant son avancement et en rapportant à la MOA. Il fait respecter les engagements en termes de contenu/qualité, coûts et délais. De même, l’AMOA animera les comités de pilotage en présence de la MOA et de la MOE.

**Question 3 :**

Le chef de projet **dirige chacune des phases du projet**. Il est chargé de convaincre les parties prenantes, de définir le périmètre du projet et, enfin, de planifier et mener à bien le projet.

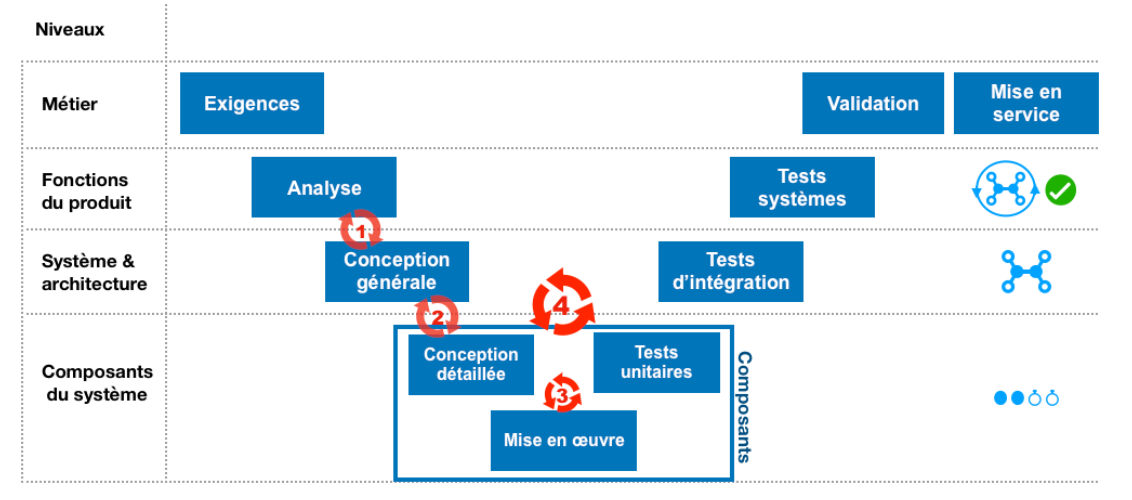
Un chef de projet détermine les tâches à effectuer pour atteindre les objectifs relatifs au projet. Cette personne forme une équipe, se munit des outils nécessaires et définit la séquence des tâches selon un échéancier précis.

Il est également en charge de **la création du budget**relatif au projet, de la **gestion des risques** et du **suivi de l'avancement**du projet. Il doit veiller à ce que les personnes travaillent sans entrave et respectent les délais prévus.

Il est important de souligner que, même si le chef de projet dirige l’équipe et fait office de leader dans le projet, il n’en est pas le patron. Le chef de projet fait accepter l'idée d'un projet au sponsor ou à une partie prenante et il leur soumet des rapports réguliers sur l'état d'avancement du projet.

En général, les chefs de projet sont des personnes très organisées, orientées résultats et passionnées. Ils sont capables de bien travailler sous pression, font preuve de leadership et savent motiver les autres.

**Question 4 :**

****

**Raisons :**

Il y a un risque important de se rendre compte au cours de la mise en œuvre que les spécifications initiales étaient incomplètes, fausses, ou irréalisables. Il y a également un risque de voir de nouvelles fonctionnalités requise par les clients (risque de [dérive des objectifs](https://fr.wikipedia.org/wiki/D%C3%A9rive_des_objectifs)), ainsi que d'autres risques évoqués dans l’ouvrage. C'est principalement pour cette raison que le cycle en V n'est pas toujours adapté à un développement logiciel. Si les projets longs durés sont adaptés à ce mode de gestion de projet, ce sont également souvent eux qui risquent de ne plus coller aux besoins qui évoluent dans le temps.

**Question 5 :**

**Les Méthodes Agiles**

### Extreme Programming (XP)

### Feature Driven Development (FDD)

### Lean Software Development

### Agile Unified Process (Agile UP or AUP)

### Crystal (Clear/Orange)

### Dynamic Systems Development Method (DSDM)

### Adaptive software development (ASD)

### Behavior driven development (BDD)

### Conception pilotée par le domaine (DDD domain-driven design)

### Test driven development (TDD)

### Rational Unified Process (RUP)

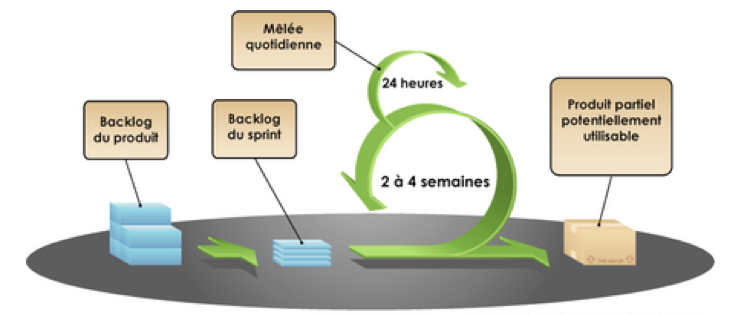
### Disciplined Agile Delivery (DAD)

### Enterprise Unified Process (EUP)

**Question 6 :**

* la primauté des personnes et des interactions sur les processus et outils.
* une préférence pour un logiciel fonctionnel plutôt qu'une documentation complète.
* une relation autre avec les clients : une collaboration permanente remplaçant une négociation contractuelle.
* une adaptation continue au changement et non le suivi rigide d'un plan.

**Question 7 :**



Scrum est une méthode de développement agile qui a été créé dans le but d’obtenir de meilleurs résultats dans les meilleurs délais. Elle se fonde sur plusieurs principes, des rôles fixes, des réunions officielles et des listes. Scrum permet de mener à bien des projets dont les besoins sont difficiles à quantifier dès le début. C’est l’une des principales forces qui ont contribué à sa popularité. Elle comporte cependant des inconvénients comme toute méthode. Scrum est adaptée aux équipes de petite taille, par conséquent des difficultés peuvent être rencontrées s’il s’agit de projets dont la taille de l’équipe est assez importante.

Scrum présente le risque de voir les fonctionnalités s’étendre indéfiniment ou « Scope Creep ». À moins qu’une date de fin soit définie de manière formelle, les parties prenantes peuvent être tentées de rajouter des fonctionnalités. Par ailleurs, si une tâche n'est pas bien définie, l'estimation des coûts et du temps du projet ne sera pas exacte. Dans ce cas, la tâche peut être répartie sur plusieurs sprints.  
  
La méthodologie Scrum est étiquetée comme « carrée ». En effet, elle ne supporte pas des modifications dans son principe. On peut lire sur le guide officiel que « les rôles, les artefacts, les événements et les règles de Scrum sont immuables et bien que l'implémentation de certaines parties de Scrum soit possible, le résultat n'est pas Scrum. Scrum n'existe que dans son intégralité et fonctionne bien comme un conteneur pour d'autres techniques, méthodologies et pratiques. »  
  
Ce manque de flexibilité peut être constaté par exemple dans la définition du PBI « Product Backlog Item », qui est sous la responsabilité du Product Owner. En cas de manque de communication, les membres de l’équipe n’auront pas la possibilité d’apporter des améliorations à la définition des produits. Le guide de Scrum ne définit pas formellement une méthode de gestion des bogues, de réduction de la dette technique.  
  
Le fait d’imposer un système de gestion de temps constitue également une contrainte. Tous les événements sont conditionnés dans le temps. Une fois qu'un Sprint commence, sa durée est fixe et ne peut être raccourcie ou allongée. Le Daily Scrum est fixé sur une durée de 15 min. Ce mode de fonctionnement ne favorise pas, une gestion de temps propre à chaque membre de l’équipe et par conséquent réduit les possibilités d’innovation et d’optimisation.  
  
Enfin la méthode SCRUM s’adapte difficilement aux outils de gestions de tâches (JIRA, TFS,…) qui imposent des interprétations très bureaucratiques de Scrum. Ce qui induit une perte de temps pour les développeurs. Vis-à-vis des développeurs, le fait de splitter les tâches en petits éléments qui peuvent théoriquement être complétés par n’importe qui dans l’équipe réduit le sentiment de fierté et d’appropriation de la réalisation des tâches. Tous ces points faibles de la méthode ont poussé Adam Ard, un professionnel de la programmation, à affirmer que Scrum est une mauvaise méthode pour la gestion de projet agile.

**Question 8 :**

Une user story est une demande fonctionnelle écrite de façon à mettre en avant les besoins utilisateurs. Elle est écrite dans un langage naturel compris par l’ensemble des acteurs du projet ou liés à celui-ci.

Pour écrire une user story agile, on suit en fait un format stricte pour que toutes les user-stories suivent les mêmes règles. Le format suivant est celui que l’on rencontre dans 90% des cas :

En tant que [persona], je souhaite [souhait] afin de [but]

As a <persona>, I want <goal/desire> so that <benefit>

## Exemple de user stories agiles

Voici quelques exemples de user-stories sur un site e-commerce :

* En tant que client je souhaite ajouter un produit dans mon panier
* (et) En tant que client je souhaite payer en carte bleue
* (ou) En tant que client je souhaite recevoir mon produit en point relais

**Question 9 :**

Découpage vertical

Cette pratique consiste à découper les user-stories sans se soucier des aspects architectures. Les user-story pourront potentiellement intervenir sur différentes couches d’architecture qui ne sont pas gérées par les mêmes compétences (C#, PHP, React/Node…).

**Question 10 :**

## Le chef de projet

Le chef de projet est la personne chargée de gérer un projet informatique ainsi que son bon déroulement avec une équipe de développeurs qu’il a en charge ; il anime cette dernière pendant toute la durée du projet informatique. On demandera à ce chef de projet d’avoir les compétences techniques et de gestion de projet pour gérer le projet informatique ; un bon relationnel est également un grand plus.

Voici les tâches principales que doit gérer le chef de projet :

* l’étude de faisabilité technique et de conception du produit (cahier des charges)
* assurer la communication avec les autres pôles extérieurs
* planifier le travail de ses collaborateurs
* sera garant de la conformité des livraisons

On pourrait rajouter d’autres tâches.

## Le Scrum Master

Le Scrum Master est dans les grandes lignes :

* + un coach
    - gardien de la bonne pratique
    - formateur sur la partie agilité
  + un facilitateur
    - * facilite l’extérieur à s’adapter au Scrum
      * écarte les perturbations
    - médecin de l’équipe
* un animateur des rituels

Dans certains contextes, le scrum master peut en effet avoir des rôles complémentaires mais en faisant attention à lui permettre de garder une neutralité sur le produit lui même.

**Dossier 2 :**