

Synthèse : Orga

Partie 1 : Le monde de l'entreprise

1.1 L'entreprise

L'entreprise est composée d'actionnaires (propriétaires du capital de l'entreprise), de dirigeants et d'autres parties prenantes. Ces parties prenantes sont les individus ou les groupes qui dépendent de l'organisation pour atteindre leurs propres buts et dont l'organisation dépend également.

Les actionnaires possèdent les dividendes les plus élevés. Les dirigeants tirent profit de leur travail. Les intérêts divergent car par exemple, les act. pourraient vouloir des dividendes plus élevés alors que l'entreprise aurait besoin d'investir. Les dirigeants, eux, pourraient exagérer les avantages en nature, rechercher la croissance à tout prix. Ils vont parfois filtrer l'information vers les actionnaires (ex : ne pas parler des problèmes).

La liquidité doit suivre la croissance (chiffre d'affaire) = ce que tu vends doit suivre ce que tu fais entrer, sinon faillite.

Au niveau des PP, les économiques sont par exemple les fournisseurs qui ont des contrats en dur avec l'entreprise et qui vont parfois même jusqu'à se localiser tout près, les concurrents qui ont établi des règles entre tous, il y a des attentes de l'un envers l'autre, distributeurs, etc.

Les PP politiques sont composées des pouvoirs publics, des régulateurs, du fisc, ... Le fisc est favorable à une non-diminution d'impôts. Le pouvoir public est favorable à de nouvelles boîtes pour la création d'emplois.

Les filiales augmentent les salaires sans pour autant envoyer tout à la maison mère.

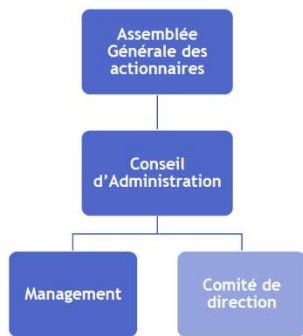
Banquiers, clients, etc.

Les PP sociétales interagissent avec l'urbanisme, les accès aux ressources, l'usure des routes à cause d'une usine, ...

Il y a également un intérêt convergent des PP. Il existe la loi « corporate governance » qui précise la responsabilité des 3 : AG, Administrateurs, et le Comité de Direction, pour qu'on puisse les suivre en justice par exemple.

Comme parallélisme avec l'Etat :

- (Population) Les électeurs élisent des députés qui forment le Parlement. Le gouvernement est géré par des ministres et le 1^{er} ministre. La Constitution de l'Etat fixe l'organisation et le fonctionnement de l'Etat.
- (AG) Les actionnaires élisent les administrateurs qui forment le Conseil d'Administration. Le management est géré par les administrateurs exécutifs et l'administrateur délégué (CEO = PDG = président directeur général). Les Statuts de l'entreprise fixe l'organisation et le fonctionnement de l'entreprise.



Le comité de direction n'est pas obligatoire.

L'assemblée générale regroupe les actionnaires de l'entreprise. Elle est souveraine (élections). Elle se préoccupe de la rentabilité du capital investi. Elle fait 5 choses :

- Nommer et révoquer les administrateurs. Chaque action vaut une voix. L'élection se fait sur proposition du CA. Vote censitaire.
- Approuve les comptes annuels.
- Décide l'attribution ou non des dividendes.
- Décide de modifier les statuts de l'entreprise (condition de présence et majorité, respectant la loi).
- Décide de modifier la structure du capital.

Elle se rassemble (assemblée ordinaire) obligatoirement une fois par an pour approuver les comptes annuels présentés par le Conseil d'Administration. Elle se rassemble (assemblée extraordinaire) exceptionnellement lorsqu'il faut modifier les statuts, augmenter le capital, voter la continuation de l'entreprise ou encore en cas de démission d'un administrateur.


Le conseil d'administration regroupe les administrateurs de l'entreprise. Il est responsable de la responsabilité juridique et du devoir de loyauté envers les actionnaires. Il peut déléguer la gestion quotidienne mais pas les décisions en matière de politique générale.

Il se préoccupe de la pérennité de l'entreprise. Il est responsable du contrôle du management et des choix stratégiques. C'est l'organe principal de gestion et de prise de décision de l'entreprise.

Il est composé de propriétaires (membres qui n'ont pas d'actions), d'employés (au jour le jour dans l'entreprise), d'indépendants (qui sont appelés car ils ont une grande capacité d'expertise) et d'autres actionnaires.

C'est le CA qui nomme le CEO (qui est là tous les jours, administrateur délégué). Celui-ci va mettre en place les règles fixées par le CA. Le CA nomme également le comité de direction et le management. Les administrateurs exécutifs ont un rôle actif dans la gestion de l'entreprise (là tous les jours).

Le CA a un président :

- Convoque le CA quand c'est nécessaire
- Définit l'ordre du jour
- Gère le débat
- Bonne gestion (corporate governance) : pas exécutif, au-dessus de la mêlée,  CEO.

Le CA s'entoure de 3 comités auxquels il délègue des prérogatives : comité de rémunérations, comité de nominations, comité d'audit.

Comité de rémunérations et nominations :

- Vérifie que l'entreprise a une politique de rémunérations cohérente.
- Composé d'administrateurs indépendants (on ne détermine pas son propre salaire).
- Organe consultatif (donne des conseils mais aucune obligation de les suivre).
- Fait des propositions au CA, qui décide des rémunérations des dirigeants de l'entreprise.

Comité d'audit :

- Vérifie la bonne application des règles comptables.
- Vérifie le bon respect des procédures.
- Convoque les réviseurs d'entreprise.
- Conseille le CA en matière de communication financière sur les résultats intermédiaires et annuels.

Le comité de direction = si les statuts l'autorisent, le CA peut déléguer tout ou une partie de ses compétences de gestion à un comité de direction, sauf bien-sûr la politique générale. Le CA délègue son pouvoir de gestion quotidienne. Il est composé d'exécutifs uniquement.

Le management est l'ensemble du personnel responsable d'une entreprise ou d'une organisation, qu'on nomme habituellement « cadre, dirigeant ou direction ». Il s'occupe :

- Du pilotage de l'action au sein d'une organisation.
- De l'organisation et la coordination des activités de l'entreprise.
- Gestion opérationnelle de l'entreprise et de la profitabilité (activités profitables à court terme) des activités.
- Met en œuvre les plans d'action qui découlent des orientations stratégiques.

Pour résumé, CA = pérennité, actionnaires = rentabilité, management = profitabilité

La taille des entreprises :

PME : micro-, petites et moyennes entreprises (TPE, PE, ME):

- Moins de 250 personnes
- ET Chiffre annuel inférieur à 50 millions d'euros.
- OU total du bilan annuel inférieur à 43 millions d'euros.



Structure de l'entreprise :

Elle vise à diviser le travail, l'organiser et assurer la coordination. Elle est représentée par un organigramme. Cet organigramme peut être dirigé de 6 manières différentes : nombre de personnes, système technologique, répartition géographique, taille, environnement ou stratégie.

On peut faire des regroupements par fonctions (même si par fonction, peut être différent d'une entreprise à l'autre), regroupements par ligne de produits, par clients (en fonction du public cible), par zones géographiques, par processus (usinage, assemblage, ...), ou une combinaison de tout ça.

Il n'y a pas d'organigramme pour les PME ou alors du self-management (organisation en fonction des compétences, initiatives, partage d'idées, chacun est son propre patron, mais attention aux excès).

Les fonctions d'interdépendance :

A compléter (57, 58).

1.2 Partie 2

Données légales :

Statuts :

- Les règles qui régissent les rapports entre les associés mais aussi ceux à l'égard des tiers
- Ils déterminent **la forme de la société** (SA, société privée à responsabilité limitée (SPRL), SPRL starter (commencer une société avec un capital moindre))
- Ils déterminent **la raison sociale** et **l'objet social**
- Ils déterminent **le siège social** (adresse)
- Ils déterminent **le capital** (montant, souscription, libération, modalité d'augmentation et réduction)
- Ils déterminent **les titres** (le nombre, la transmission, les périodes où c'est ouvert, elles sont là pour augmenter le capital. On réserve une part pour les actionnaires pour garder la même proportion du capital, une part en bourse (dont peut-être une part pour ses employés peut-être sous forme de warrant).) Le warrant est une promesse qu'on peut acheter une action quand on veut qui nous est de toutes façons réservées. L'avantage des warrant est que ça fidélise le personnel (ils ont plus intérêt à rester).
- Ils déterminent **la représentation** (la désignation des organes de gestion, leurs pouvoirs et la manière de les exercer)
- Ils déterminent **les assemblées générales** (les règles, l'exercice social, obligations comptables, répartition des bénéfices)
- Ils déterminent **les règles de modification des statuts**.

Qu'est-ce que la raison sociale ? C'est le nom officiel de la personne morale, par exemple SPRL. Elle est unique et est reprise dans les statuts et publiée au Moniteur Belge. Elle peut être différente du nom commercial.

Qu'est-ce que l'objet social ? C'est l'ensemble des activités que la société se propose de développer. C'est repris également dans les statuts et publié au Moniteur Belge. C'est FONDAMENTAL. On ne peut pas mettre n'importe quoi, il faut des diplômes, des valorisations d'acquis ou encore une mise à niveau pour la reconnaissance de certains métiers (permaculture, etc.).

Projet d'entreprise :

La base d'un projet d'entreprise est la **MISSION**. C'est l'intention fondamentale du projet, sa raison d'être. C'est le métier de l'entreprise, ce qu'elle produit. =**contribution de l'entreprise à la société**.



Cette mission est un texte de référence pour l'extérieur mais aussi l'intérieur. Il doit être durable et stable dans le temps. C'est un repère stable. Attention, la façon d'orienter sa mission peut être différente selon la société. Il faut faire attention à ne pas trop la « teinter ». La mission est importante pour les clients.

La deuxième chose sont les **VALEURS**. Ce sont les convictions, les principes, la façon d'agir. =**la culture d'entreprise**. Elles ne doivent pas varier en fonction des circonstances. Elles orientent les attitudes et les comportements du personnel. Le but est de réduire les conflits. L'ensemble des valeurs façonne notre culture. Ça peut être le respect (employé et société civile), l'engagement (clients et partenaires), la compétitivité (rentabilité et concurrence). La culture d'entreprise est par exemple la disposition des bureaux, le nombre de réunions, le partage et l'enthousiasme, etc.

Le troisième pilier est la **VISION**. =la perception de l'avenir. C'est la vue à long terme. Il faut anticiper et prévoir (environnement économique, concurrents, ressources disponibles, ...). Cette vision vient du CA qui a comme objectif la pérennité de l'entreprise. Elle est mobilisatrice pour tous les collaborateurs. Ce sont également les réalisations à accomplir. Elle doit nous faire savoir où l'on va. Tout le monde doit la connaître (être n°2 par exemple).

Ces 3 premiers piliers sont risqués si les parties prenantes n'y adhèrent pas.

C'est pour ça que le suivant sont les **OBJECTIFS GÉNÉRAUX**. Ils vont décliner la vision en objectifs opérationnels. Ils sont divisés en catégories (développement, marketing, commerciaux, RH, etc.). C'est l'énoncé des résultats qui doivent être atteints. Ça peut par être financier, commercial, environnemental ou sociétal.

On finit avec les **OBJECTIFS DE CHACUN**. C'est faire comprendre à chacun en quoi son activité permet d'atteindre les objectifs exprimés par la direction.

Responsabilité Sociétale des Entreprises :

RSE = CSR. Ce sont les pratiques d'affaires durables, de meilleures pratiques, plus responsables. Les parties prenantes sont L'AG, les administrateurs et le comité de direction. Elles suivent seulement la loi de Corporate Gouvernance. Les autres parties prenantes n'ont aucune loi, elles sont conseillées alors de suivre la RSE.

Elle désigne l'intégration volontaire d'objectifs sociaux et environnementaux dans les activités commerciales des entreprises, et dans leurs relations avec leurs partenaires. C'est bien évidemment aussi une technique de marketing.

Quelles-sont ces parties prenantes ? Les pouvoirs publics, les gouvernements, les communes, les fournisseurs, les distributeurs, les clients, l'association consommateurs, les actionnaires, l'ONG, les banques, les employés, les syndicats et les associations professionnelles.

Leur intérêt à suivre cette loi est d'intégrer leurs attentes et leurs intérêts. Par exemple, ne pas regarder que la rentabilité financière, conserver les activités déficitaires (pour garder le quota des handicapés). Ne pas vendre des produits qui ont un coût social (maladies professionnelles dues à un produit). Dépenser pour la collectivité (pharmacie qui incinère les vieux médicaments).

Une autre définition de la RSE est l'engagement des organisations à se comporter de manière éthique et à contribuer au développement économique tout en améliorant la qualité de vie de leurs employés, de leur famille et de la société au sens large.

Voici les objectifs :

- Inclusion des groupes défavorisés dans le marché du travail
- Formation tout au long de la vie
- Amélioration de la santé publique
- Innovation en matière sociale et environnementale
- Réduction de la pollution + emploi rationnel des ressources naturelles
- (+valeurs européennes, droits de l'homme, etc.)

Les entreprises sont en partie responsables :

- De la protection sociale des employés (mutualité, congés maladies, aide aux conjoints, etc.),
- Des conditions de travail (environnement, normes de sécurité),

- Des aspects écologiques (pollution, préservation de l'énergie),
- Des produits (maladies professionnelles),
- Des fournisseurs (procédure d'achat équitable ?),
- Des emplois (les minorités, préserver l'emploi en cas de situation difficile),
- Des droits de l'Homme (travail des enfants, liberté syndicale)

En bref, la RSE réduit le risque de réactions négatives de la part des parties prenantes. Elle attire et conserve un personnel loyal et motivé. Elle contribue à l'avantage concurrentiel : situations mutuellement bénéfiques.

L'entrepreneuriat social :

C'est une entreprise qui poursuit une mission sociale tout en assurant sa visibilité financière au travers d'une activité économique rentable (micro-crédits, ou ateliers protégés (handicapés) par exemple). Ça n'est pas obligé d'être rentable.

La différence entre la RSE et l'entrepreneuriat social est que la RSE à une prise de conscience moral ou un geste moral ou environnemental, elle apporte du social à l'économie. L'entrepreneuriat social a, lui, un besoin d'innovation dans le secteur social, il injecte de l'économie dans le social.

Au niveau de l'éthique personnel, si ça n'est pas légal, on dit non. Si ça n'est pas en phase avec les valeurs de l'entreprise, on dit non. Si ça n'est pas en phase avec vos valeurs, il faut s'interroger. A voir si on oserait tout avouer devant la presse.

1.3 L'entreprise et le modèle social belge

Système de concertation sociale :

C'est un système **institutionnel** de négociation. Il est présent à différents niveaux entre les partenaires sociaux : organisations patronales (représentant les employeurs) et organisations syndicales (représentant les travailleurs). Ce système permet de conclure les **Conventions collectives de travail (CCT)**.

Conventions Collectives de Travail :

C'est un accord conclu entre une ou plusieurs organisations syndicales et une ou plusieurs organisations patronales ou un ou plusieurs employeurs. Il fixe les relations individuelles et collectives de travail entre employeurs et travailleurs d'entreprises ou d'une branche d'activité. Il règle les droits et devoirs des parties contractantes.

L'objectif est de donner un cadre qui est toujours en rapport avec le travail. Le contrat de travail individuel ne peut déroger à ce document. L'employeur ne peut pas non plus priver l'employé de droits qu'il a obtenu en vertu d'une CCT. Ça vient structurer les relations entre les patrons et les employés.

Différents niveaux :

Chaque niveau inférieur respecte le cadre de celles du dessus.

1) Niveau interprofessionnel (CCT intersectorielles) :

Il est composé d'un **accord interprofessionnel**. Aussi appelé accord-programme ou accord-cadre, il est conclu tous les 2 ans par les représentants des partenaires sociaux du secteur privé. Il est élaboré au sein du **groupe des 10**. Il s'applique à tous les secteurs. Il contient des dispositions tels que les

formations, le SMMM, évolution salariale, les réductions de charges, les revenus de remplacement, etc. Il trace également un **cadre pour les négociations dans les secteurs**.

Le **groupe des 10 (ou groupe des partenaires sociaux)** est composé de 11 membres : le **banc syndical** composé de 2 représentants de la CSC / ACV, 2 de la FGTB / ABVV et 1 de la CGLSB / ACLVB. Et le **banc patronal** composé de 2 + 1 de la FEB (fédération des entreprises en Belgique, +1 pour le président du groupe) / VBO, 1 de l'Unizo (fédération pour les indépendants), 1 pour l'UCM (Union des classes moyennes) et 1 du Boerenbond.

Il fixe les lignes de force en ce qui concerne les **grands thèmes socio-économiques** et insiste sur le fait qu'il veut être entendu et **impliqué lorsque les choix politiques** doivent être faits en la matière. Il décide par exemple le taux d'augmentation des salaires, le budget bien-être, comment on alloue le budget, etc.

Dossiers concertations sociales

- formation des salaires pour une période de 2 ans et la question éventuelle marge salariale.
- répartition de l'enveloppe bien-être qui doit permettre de relever les allocations les plus basses (petites pensions) et les revenus de remplacement les plus bas (allocation maladie, accident, etc.).
- convention collective

Le niveau est aussi siégé par le Conseil National du Travail (CNT) et le Conseil pour la prévention et la protection au travail (PPT), qui donnent principalement leur avis.

2) CCT sectorielles :

Basées sur les commissions paritaires qui sont mises en place pour négocier les CTT dans chaque secteur d'activité. Les résultats concernent aussi bien les **conditions de travail** que le **salaire** ou encore la **gestion de la paix sociale**. Voici leurs missions : conclure des conventions collectives de travail, prévenir ou régler des conflits sociaux, conseiller le gouvernement, le Conseil national du travail ou le Conseil central de l'économie, accomplir chaque mission qui leur est confiée par une loi.

Qu'est-ce que la **paix sociale** ? Lors de la conclusion des conventions collectives de travail, les représentants du patronat et ceux des syndicats s'engagent, pendant la durée de validité de cette CTT, à respecter les dispositions contenues dans l'accord. Cela permet de garantir la paix sur le plan social.

3) CCT d'entreprise :

Il y a 2 organes dans une entreprise : le **comité pour la prévention et la protection au travail** et le **conseil d'entreprise**.

Le CPPT prend des initiatives afin d'améliorer le bien-être des travailleurs dans les entreprises. Ça concerne la participation directe des travailleurs dans le cadre du bien-être au travail. Les dates de réunion varient en fonction du nombre de travailleurs : le comité doit être institué dans toutes les entreprises d'au moins 50 travailleurs en moyenne. Si ça n'est pas le cas, les tâches et compétences de cet organe de concertation sont transférée à la **Délégation syndicale**. Si celle-ci n'existe pas non plus, alors on fait appel aux travailleurs-mêmes.

Sa mission est de rechercher et proposer tous les moyens et de contribuer activement à **tout ce qui est entrepris pour favoriser le bien-être des travailleurs lors de l'exécution de leur travail**.

Attention, ils ne prennent pas de décision, mais on a besoin d'eux pour des compétences d'avis, de propositions et de droit à l'information.

Le conseil d'entreprise est un organe de consultation entre l'employeur et les représentants des travailleurs. Il est bipartite, c'est-à-dire qu'il est composé des représentants des travailleurs, élus lors des élections sociales (tous les 4 ans) et du chef d'entreprise et de ses délégués désignés par lui.

Sa mission est de recevoir des sur la situation de l'entreprise (économiques, financières, en matière d'emploi, etc.). C'est une obligation de donner ces informations.

Il doit également élaborer ou modifier le **règlement de travail** en vigueur dans l'entreprise. Mais aussi gérer les **œuvres sociales** instituées dans l'entreprise pour le bien-être du personnel. Et encore **donner des avis et formuler des suggestions** quant au fonctionnement de l'entreprise. C'est par exemple lui qui déciderait s'il faut mettre des terrains de sport, etc.

Il doit aussi **vérifier le respect des législations qui protègent le personnel** :

- Application de dispositions légales ou réglementaires
- Application de dispositions résultant des conventions collectives
- Application du règlement de travail
- Mesures collectives d'ordre social prises unilatéralement par l'employeur

Chapitre 2 : Les processus

Qu'est-ce que c'est ?

C'est l'ensemble d'activités organisées dans le temps. Le processus prend un input, y ajoute de la valeur et produit un output vers un client. Il produit un résultat précis et mesurable. Précisons que tant les entrées que les sorties sont clairement identifiables.

Il peut être exécuté plusieurs fois. Un processus peut faire partie d'un autre processus et il peut également contenir plusieurs processus.

Comment identifier les processus de l'entreprise ?

Les processus de réalisation sont de développer (un logiciel), fabriquer (des voitures), vendre (le logiciel ou les voitures), concevoir (un nouveau produit), acheter (les matières premières), etc.

Les processus support sont de gérer le personnel (RH...), gérer le système IT, etc.

Les processus de management sont de diriger les équipes, auditer, contrôler, définir la stratégie, planifier le travail, analyser les statistiques, produire les documents du système qualité, etc.

Pourquoi se centrer sur les processus ?

Car ça permet de se centrer sur le client, fournir une vue systématique des activités de l'entreprise et permettre de comprendre comment les inputs se transforment en output.

Lorsqu'ils sont décrits, les gains attendus sont de comprendre comment l'entreprise pourrait fonctionner au mieux, fournir des mesures de gaspillages, pertes et déchets, comprendre comment les erreurs se produisent et développer une méthode pour les corriger, développer un système de mesure pour toutes les activités et enfin permettre de prévoir et de contrôler le changement.

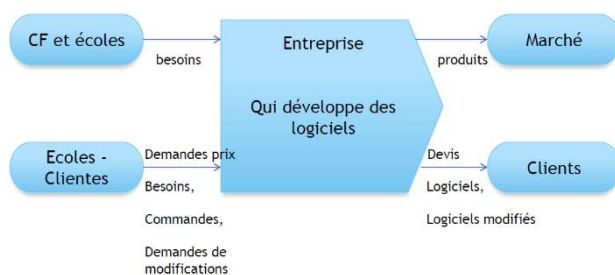
Décrire leurs processus oblige les entreprises à documenter ce qu'elles font, à les **formaliser**. Une méthode de travail est nécessaire pour mieux s'organiser lorsque l'on est plusieurs à travailler. On a par exemple besoin de définir les processus des achats car ils sont répétitifs et on souhaite que ça se fasse automatiquement. Cela se fait par étape, on commence par l'achat du café dans une petite entreprise, mais dès qu'elle grandit, on a besoin de plus et on doit gagner du temps.

Illustration

Prenons l'exemple que 4 personnes développent un logiciel de stage et veulent le vendre à grande échelle. L'actionnariat est fermé et le capital se structure comme suit : fonds propres limités, principalement emprunts. **Pour une simple demande d'achat de PC, il faut définir le type pour tous les employés suivants, sélectionner un fournisseur en fonction du prix, de la durabilité, des back up dans la cave, etc. Il faut également qu'il y ait du respect de la part du fournisseur. Dans le contrat avec le fournisseur, il faut qu'il y ait automatiquement une commande d'achat dès qu'il y a un nouvel employé.**

Dans la vue du processus, en input, il faut les infos et demandes du marché (par exemple au minimum n années de stage pendant les études), et des clients (la demande de choses particulières). En output, l'entreprise fournit de l'information et le logiciel.

La Communauté Française et les écoles ont besoin d'une entreprise qui développe des logiciels. Les écoles clientes demande le prix, définissent les besoins, font les commandes et demandent des modifications à cette entreprise. Cette entreprise produit sur le marché, et elle envoie des devis, les logiciels et les logiciels demandés aux clients. C'est le système d'input et d'output.



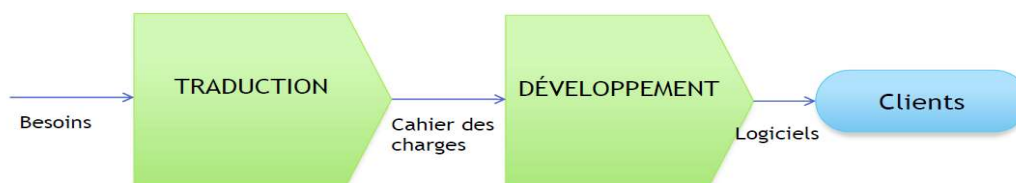
Il y a différentes casquettes :

- Celle du marketing qui analyse les besoins du marché et définit les produits
- Celle du commercial qui prospecte les clients, reçoit les demandes de prix et les commandes
- Celle du comptable/financier qui facture les

clients, travaille avec les banques, ...

- Etc.

Ces services travaillent selon les processus qui font partie du processus « unique » décrivant l'entreprise. Tant les demandes du marché que les demandes des clients sont traduites sous forme d'un cahier des charges. Il y a également des informaticiens qui travaillent sur le processus principal : le processus de développement de logiciel.



Scénario : l'entreprise évolue

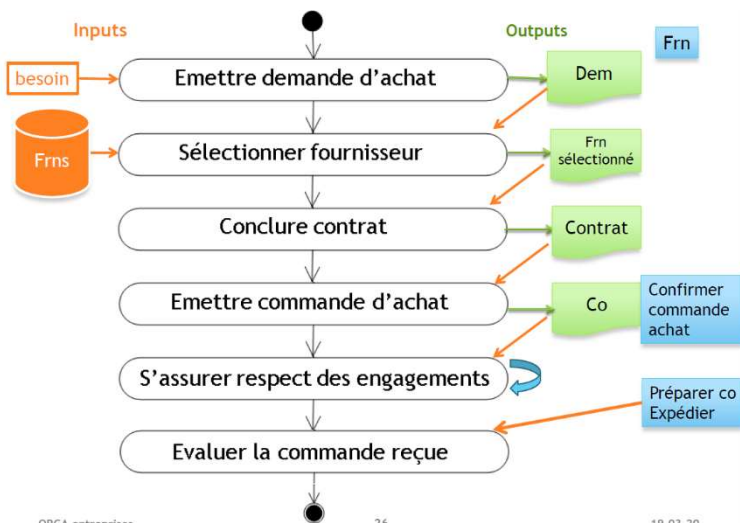
Il faudra développer et stabiliser le logiciel pendant la 1^{ère} année, contacter les écoles supérieures pendant ce temps. Ensuite déployer le logiciel pendant la 2^{ème} année dans les 7 écoles, engager 2 développeurs la 2^{ème} année, développer les premiers contrats « clé sur porte » dans le courant de la 2^{ème} année. Ensuite, encore des engagements, la livraison des premiers logiciels « clé sur porte ». Les bénéfices vont grimper. La croissance de la société va être rapide. Il va falloir définir des méthodes de travail. On aura besoin d'une infrastructure solide. Les processus d'achats pour les pc, les imprimantes, les serveurs, les licences, les papiers-entête, les consommables devront être définir **(plus haut)**.

L'entreprise devra définir ses processus, savoir lesquels détailler, etc.

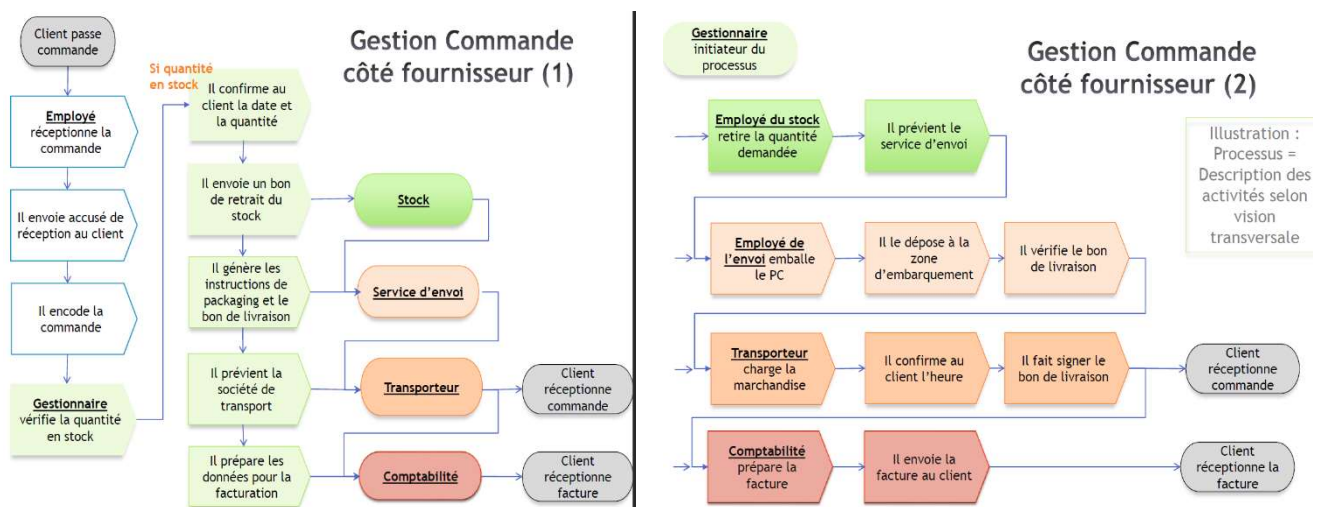
Les processus d'achats :

Les étapes : émettre une demande d'achat (la spécification du produit) – sélectionner le fournisseur – conclure un contrat – émettre une commande d'achat – s'assurer du respect des engagements – évaluer la commande reçue.

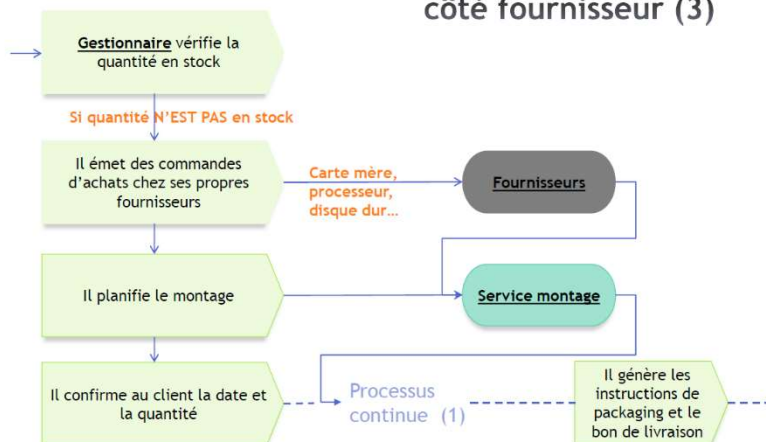
On remarque que les premiers outputs deviennent les suivant input :



Les processus de gestion de la commande (vente) :



Gestion Commande côté fournisseur (3)



Tout un processus contenant lui-même un ou plusieurs sous-processus.

Pharmacie :

Dans une pharmacie, le processus de stock est complexe car tout n'est pas en vente libre, tout ne se conserve pas de la même manière et surtout, il y a des dates de péremption. Ce sont les fournisseurs qui doivent ajouter leurs produits dans la base de données. Le pharmacien sait habituellement ce qu'il vend. Pour chaque produit, on connaît tout un tas de données. Lors d'une vente, le stock est décrémenté. Si le seuil de commande est atteint, une ligne de commande est automatiquement créée avec un certain nombre d'exemplaires. Ce nombre d'exemplaires varie en fonction des périodes (saisons, pharmacie de garde, épidémie, etc.). Des commandes sont envoyées automatiquement plusieurs fois par jour.

Pour gérer les produits périmés, chacune des dates de péremption est encodée dans le système dès la réception. Une fois par mois, on sort ceux étant périmés du stock. Il faut également tenir compte des produits qui se prennent pendant x mois et qui périssent dans $n < x$ mois.

Pour représenter un processus, on doit répondre à plusieurs questions :

- Est-ce un seul processus ?
- Y a-t-il des parties intégrées dans un autre processus ? Qui en est responsable ?
- Y a-t-il des processus qui se lancent automatiquement ? Qui en est responsable ?
- Est-ce un processus important ? Quel suivi doit-on installer ?
- ...

Autres processus :

Il existe pleins d'autres processus tels que la vente, le suivi des plaintes, la maintenance planifiée, le GHR (contrat employés, paie, planification des congés), etc.

Pour une bonne représentation, les **flux** et les **interactions** doivent être mises en évidence. On utilise aussi parfois un diagramme de flux mais ça ne représente pas vraiment le processus car on ne voit pas l'ordre des activités.

Automatisation de la gestion des processus :

Il est certain qu'on a besoin d'une norme pour dessiner les processus. Il faut des modèles de documents. Pour chaque processus, il faut la définition des rôles et responsabilités à chaque étape, la définition des inputs et outputs ainsi que les points de contrôle (quality gates). Une norme peut se trouver sur le Business Process Model and Notion : <http://www.bpmn.org>.

Chapitre 4 : Contrat de travail

Un **contrat de travail** est un contrat par lequel une personne s'engage à travailler moyennant rémunération sous l'autorité d'une personne qui l'emploie. Cette notion d'autorité est à prendre avec attention. On dépend et on doit accepter l'autorité de 3 personnes en particulier : le gestionnaire d'équipe, le directeur du département et le CEO.

Le travailleur a pour obligation « d'agir conformément aux ordres et aux instructions qui lui sont données par l'employeur, ses mandataires et préposés, en vue de l'exécution du contrat ».

Lien de subordination juridique :

- L'employeur a le pouvoir de donner des ordres et des directives, d'en contrôler l'exécution et de sanctionner les manquements.
- Le travailleur a l'obligation d'agir conformément aux ordres donnés.
- En contrepartie, l'employé a des droits.

Il comporte à :

- Volonté des parties, exprimée dans le contrat.
- Être soumis à des horaires (fixes, variables ou semi-variables), à une organisation de temps de travail.
- Travailler dans un lieu déterminé par l'employeur.
- Recevoir une définition précise des tâches à accomplir, des instructions et des décisions d'un supérieur hiérarchique -> puisqu'on peut être sanctionné (jusqu'à l'exclusion), on doit nous dire ce qu'on doit faire.
- Être soumis à un contrôle hiérarchique.
- Recevoir le matériel ou les outils nécessaires à l'exécution du travail (fournis par l'employeur).

En contrepartie, le lien de subordination pour les travailleurs (salariés) a comme avantages :

- Salaire dont le minimum est déterminé par la loi ou les conventions collectives.
- Droit au respect de sa vie personnelle sur le lieu de travail (mais on peut nous limiter par exemple pour appeler à l'étranger la famille sur le téléphone du bureau, etc.).
- L'employeur est tenu de respecter une réglementation en matière d'hygiène et de sécurité.
- La rupture du contrat de travail est bien encadrée, qu'il s'agisse d'un CDD ou d'un CDI.
- Le salarié a des interlocuteurs dans et hors de l'entreprise pour l'aider à défendre et faire respecter ses droits (on peut se faire aider -> syndicats).

CDI :

Contrat par défaut si rien n'est stipulé. Il n'y a pas de limite de temps. On est engagé en principe pour toujours. Chaque partie peut rompre le contrat à tout moment, moyennant respect de la loi.

CDD :

La durée est fixée à l'avance. On ne peut normalement pas l'interrompre et il est interdit d'en conclure plusieurs successivement sauf :

- Lorsque la nature du travail le justifie (prolongation de saison touristique)
- Lorsque des raisons légitimes peuvent le justifier. Ça doit être démontré par l'employeur.
- Maximum 4 contrats successifs ou une durée totale \leq à 3 ans (avec l'accord du contrôle des lois sociales).

Seul les CDI temps plein n'ont pas besoin d'être écrits pour être valables.

Voici ce que doit-il contenir : une date de début et éventuellement de fin, le travail à exécuter (la fonction), le lieu d'exécution, l'horaire de travail, la rémunération convenue et mode de calcul, et les autres conditions de travail comme par exemple la clause de non-concurrence.

Clause de non-concurrence :

Lorsque l'on quitte un travail, on ne peut pas aller travailler dans une société concurrente mais dans ce cas-là, l'entreprise doit nous dédommager. Par exemple, si l'entreprise B nous propose 3000€ par mois, l'entreprise A, si elle veut nous empêcher d'y aller, doit nous déboursier cette somme là durant la durée de cette clause. C'est donc limité dans le temps. On va donc être indemnisé. On est dans l'obligation d'avertir l'entreprise A qu'on veut aller dans la B, même si tu crées ta propre société. L'entreprise A peut oublier cette clause si elle n'est pas d'accord de payer le montant de l'indemnité.

Période d'essai

Si cette période est ajoutée dans le contrat, la clause sera nulle. Il y a une période de préavis dans le cas où survient un licenciement ou une démission dans les 3 premiers mois, qui dure 2 semaines. Sauf les contrats d'occupation d'étudiants et les contrats de travail intérimaire. Depuis 2014, elle n'existe plus pour les salariés.

Obligations du salarié

Il se doit d'exécuter son travail avec soin, probité et conscience, au temps, aux lieux et aux conditions convenues. Il doit agir conformément aux ordres et instructions de l'employeur et de ses délégués

(selon le règlement de travail -> fournir une justification aux absences, respect des conditions d'utilisation des outils de communication, obligation de se soumettre à des contrôles médicaux (maladie professionnelle, yeux, chaise, c'est dans l'intérêt de l'employé), le port obligatoire de vêtements spécifiques). Ce sont des exemples.

Il doit s'abstenir durant son contrat mais aussi après cessation de celui-ci de divulguer des secrets de fabrication ou d'affaires, ainsi que le secret de toute affaire confidentielle. Il doit s'abstenir de se livrer ou de coopérer à tout acte de concurrence déloyale à son employeur.

Il peut mener une concurrence loyale à la fin du contrat sauf si clause de non-concurrence.

On ne peut pas débaucher un personnel concurrent, mais les employés peuvent partir d'eux-mêmes. Tout ça n'est pas dans le contrat, c'est une obligation. Sauf la clause de non-concurrence.

Par concurrence déloyale j'entends dénigrer une entreprise concurrente, se faire passer pour son concurrent, désorganiser le commerce de son concurrent, etc.

Le salarié doit veiller à sa sécurité, à celle de ses collègues, de l'employeur ou de tiers. Et il doit finalement restituer en bon état à l'employeur les instruments de travail et les matières premières qui lui ont été confiés.

Obligations de l'employeur

Il doit assurer le respect des convenances et des bonnes mœurs (harcèlement, etc.). Il doit faire travailler le travailleur dans les conditions, aux temps et lieux convenus, en mettant à sa disposition l'aide, les instruments et les matières nécessaires à l'accomplissement du travail. Veiller à ce que le travail s'accomplisse dans des conditions convenables (sécurité du travailleur, santé, premiers secours) -> parrain. Il faut toujours quelqu'un qui assure les gestes de premiers secours, un référent.

Il doit payer la rémunération aux conditions, aux temps et aux lieux convenus. Il doit consacrer l'attention et les soins nécessaires à l'accueil des travailleurs et en particulier des jeunes. Il doit apporter les soins à la conservation des instruments de travail appartenant au travailleur et des effets personnels que celui-ci doit mettre en dépôt. Il ne peut retenir ses instruments de travail ou ses effets.

Il se doit de donner au travailleur le temps nécessaire pour remplir les devoirs de son culte et les obligations civiques résultant de la loi. Il doit fournir au travailleur un logement convenable et une nourriture saine et suffisante dans le cas où il s'est engagé à le loger et le nourrir.

Obligation d'accord en cas de modification du contrat

Le contrat doit être exécuté selon les modalités convenues par les parties. Le travailleur ou l'employeur doit avoir l'accord de l'autre pour y apporter des modifications (sauf clauses prévues à cet effet). C'est pour cela qu'il faut bien lire un contrat avant de le signer pour savoir à quoi on s'engage.

Rupture de contrat

- **avec préavis** : Il faut prévenir l'autre partie à l'avance grâce au délai de préavis, la durée étant fixée par la loi. Il faut une notification écrite obligatoire. Il faut obligatoirement mentionner la date de début du préavis et la durée.

- **avec indemnité compensatoire de préavis** : La rupture est immédiate. On dédommage les dus en cas de rupture immédiate. On va donc payer le salaire pour la longueur du préavis directement en une fois.

- **pour faute grave** : Lorsque la faute rend définitivement et immédiatement impossible toute collaboration professionnelle entre l'employeur et l'employé. On nomme les absences injustifiées, les vols, l'insubordination, les agressions, les actes de concurrence déloyale, etc. Rater un contrat avec un client ne sera donc pas une faute mais une erreur.

- **pour cas de force majeure** : Une entreprise qui brûle, l'arrêt de l'entreprise, le coronavirus ou pas.

Délais de préavis

Plus l'ancienneté est longue, plus le délai sera grand. Il sera plus grand en cas de licenciement. En cas de démission, il sera de moitié par rapport à ceux de licenciement. Toujours avec la démission, il sera de maximum 13 semaines. Il y a bien sûr des exceptions. Ce sont des préavis sur CDI. Si la société ne peut pas payer l'entièreté du préavis, il existe un fond de l'Etat mais il ne donnera jamais la totalité.

Chapitre 4 : Rémunérations

Petit rappel : un contrat de travail possède 3 caractéristiques : c'est un travail, il se fait contre rémunération et il se fait sous l'autorité de quelqu'un.

Définition : c'est la contrepartie du travail effectué par le travailleur. Elle comprend le salaire en espèce, les éventuels avantages en nature, les avantages de toute nature et les primes et indemnités. Elle ne comprend pas le remboursement de frais professionnels.

Principes : L'employeur et le travailleur sont en principe libres de fixer le montant de la rémunération, c'est une négociation. Mais en pratique, ça tient compte du RMMMG (fixé par les accords interprofessionnels), de la norme salariale (fixée tous les 2 ans et déterminant la marge d'augmentation des coûts salariaux), du barème fixé par la convention collective sectorielle en commission paritaire (représentants du patronat et des syndicats) et enfin du barème de l'entreprise. Le salaire doit respecter toutes ces contraintes.

Lors de la fixation de la rémunération, il n'y a pas de lien de subordination (autorité).

Au niveau des principes, la rémunération doit être déterminée (montant précis) ou déterminable (salaire à la pièce, salaire horaire, rémunération à la commission, ...).

Norme salariale :

- Fixée par les accords interprofessionnels.
- Elle détermine la marge d'augmentation des coûts salariaux. Elle est fixée tous les 2 ans.
- Elle crée la possibilité d'adapter de façon préventive l'évolution des coûts salariaux en Belgique à l'évolution attendue des coûts salariaux chez nos principaux partenaires commerciaux (Allemagne, Pays-Bas et France).
- Elle sert d'encadrement pour la concertation sur les salaires qui a lieu tous les 2 ans.
- La loi sur la norme salariale prévoit que les indexations et les augmentations barémiques sont toujours garanties.
- La loi sur la norme salariale s'applique aux employeurs et aux travailleurs du secteur privé.

En 2017-2018, l'augmentation maximale pouvait être de 1,1%.

Le RMMMG : Il s'agit du revenu minimum mensuel moyen garanti. Il est fixé, via les conventions collectives, par le Conseil national du travail (CNT). C'est la limite inférieure absolue pour la rémunération. Ça n'est pas exactement équivalent à un salaire minimum mensuel car il comprend certaines sommes payées dans le courant de l'année. Par exemple, une prime de fin d'année et/ou un treizième mois, sont pris en compte pour s'assurer du respect du RMMMG. Pour illustrer ça, il faut le voir comme tel :

La somme des 12 rémunérations mensuelles + la prime de fin d'année + les éventuels autres avantages qui ne sont pas exclus par la CCT (heures supplémentaires, etc.). Cette somme est supérieure à 12xRMMMG.

Il s'agit d'un minimum obligatoire pour l'ensemble des secteurs. Plusieurs CCT, dont celle du 28 mars 2013, ont modifiés progressivement les conditions pour obtenir le RMMMG à partir de 18 ans.

Si on a moins de 18 ans, il y a un pourcentage dégressif du RMMMG. Si on a plus de 18 ans, alors il est évalué en fonction de notre âge et le nombre d'années d'ancienneté dans l'entreprise.

Il s'agit du premier cadre légal de fixation de salaire.

Salaire minimum dans l'UE : il y a d'énormes disparités entre les pays au niveau du salaire minimum. Certains sont plus hauts, d'autres plus bas, certains se basent sur les conventions collectives et d'autres n'en ont tout simplement pas.

Droit / Obligation : Au niveau du décompte / de la fiche de paie, l'employé a le droit de connaître le mode de calcul de sa rémunération et les retenues opérées sur les sommes qu'il a gagnées. L'employeur a donc l'obligation de remettre un décompte au travailleur.

Avantages en nature : Ce qui peut être fourni à titre de rémunération en nature. Ils viennent compléter le salaire en espèce. Par exemple le logement, le gaz, l'électricité, l'eau, le chauffage, la nourriture fournie par l'employeur et consommée sur le lieu de travail, la jouissance d'un terrain, etc.

Avantages de toute nature : ce sont les avantages particuliers consentis par l'employeur, à un travailleur, en plus de la rémunération habituelle, en raison ou à l'occasion de l'exercice de l'activité professionnelle. Ça peut être la disposition gratuite d'un véhicule de société, des primes d'assurances que l'employeur paie au profit du travailleur (assurance individuelle contre les accidents de la vie privée), la mise à disposition d'un GSM ou d'un PC pour usage privé, etc.

Tous ces avantages seront également soumis au calcul des cotisations sociales et à la retenue du précompte professionnel.

Salaire :

Le salaire de base est le salaire brut de l'employé, avant toute contribution de type fiscal ou social. Le salaire net est le salaire payé.

Il faut toujours négocier un salaire brut puisque le salaire net varie en fonction de notre situation familiale.

Salaire brut – ONSS = salaire imposable

L'ONSS est l'Office Nationale de Sécurité Sociale.

Salaire imposable – précompte professionnel – contributions spéciales pour la sécurité sociale - ... + ... = salaire net.

Le salaire imposable est le salaire sur lequel l'impôt est calculé. On ne connaît pas l'impôt au départ, donc on enlève un précompte professionnel, qui est une avance sur l'impôt.

Le salaire peut être fixe (le même chaque mois), variable (fixé en fonction de la tâche effectuée) ou semi-variable (il y a un fixe et une partie variable liée aux résultats).

Le salaire peut être mensuel, hebdomadaire, journalier, par heure ou à la tâche.

Comment calculer un salaire journalier au départ d'un salaire mensuel : Puisque le nombre de jours est variable, on va passer au trimestre (13 semaines).

S.J. = S.M x 3 / 16 / régime hebdo.

Cotisations sociales : c'est la contribution ONSS, c'est la contribution à la sécurité sociale de l'EMPLOYÉ.

Le montant est de 13,07% du salaire brut. Ça concerne tous les secteurs : allocations en cas de maladie, allocations de chômage, allocations familiales, assurances soins de santé et indemnités, pensions, maladies professionnelles, accidents du travail, etc. Tous ces budgets viennent de nos cotisations et celles de l'employeur (cotisations sociales patronales). Ça a pour but la solidarité entre les personnes.

Précompte Professionnel : c'est un paiement anticipé des impôts des personnes physiques. Il est calculé sur base du salaire imposable. Il prend en compte la situation familiale de l'employé. C'est pourquoi le calcul est complexe et donc souvent fait par le secrétariat social de l'employeur.

Calculs :

En 2019, le salaire moyen mensuel brut pour un développeur était de 2618€. Mais ces chiffres ne sont pas très fiables car l'échantillon est petit (612).

En 2020, il est de 3156€. Ça s'explique par la forte demande pour des informaticiens. Mais il faut se méfier car il y a cette fois-ci 376 observations.

Le salaire évolue aussi différemment d'une région à l'autre. On sera mieux payé à Bruxelles ou à Anvers qu'au Luxembourg.

Voici le site pour calculer : <https://www.fgtb.be/calcul-salaire-brut-net>

Revenu brut mensuel*	
<input type="text" value="2200"/>	Brut - cotisations sociales personnelles + bonus à l'emploi pour les bas salaires = revenu brut imposable
Cotisations sociales personnelles:	<input type="text" value="287.54"/>
Bonus à l'emploi pour les bas salaires:	<input type="text" value="79.11"/>
Revenu brut imposable:	<input type="text" value="1991.57"/>
Précompte Professionnel:	<input type="text" value="297.72"/>
Réduction du précompte professionnel pour les bas salaires:	<input type="text" value="26.22"/>
Cotisation spéciale pour la sécurité sociale:	<input type="text" value="18.71"/>
Revenu net:	<input type="text" value="1701.36"/>

Salaire net – situation familiale :

De personne isolée à personne mariée 1 revenu : précompte professionnel diminue fortement.

De personne isolée à personne mariée 2 revenus : précompte professionnel augmente un petit peu.

De personne isolée à personne mariée 2 revenus et 1 enfant à charge : précompte professionnel légèrement moindre.

On peut aller sur le site de la fgtb puisque c'est un syndicat qui fait partie du groupe des 10. On peut aller sur d'autres sites de syndicats et sur les sites des associations patronales. Fiable.

Salaire net – comparaison : comparons maintenant pour une personne isolée en fonction du revenu brut souhaité.

S'il on augmente de 500€ le brut, le net n'augmentera pas autant, pourquoi ?

- les cotisations sociales personnelles en pourcentage augmentent
- le bonus à l'emploi est moindre
- le précompte professionnel augmente puisqu'il ne se trouve plus dans la même tranche
- la cotisation spéciale pour la sécurité sociale augmente.

Le net augmentera mais moins que le brut.

Salaire annuel brut :

C'est la somme des 12 mois x salaire mensuel + 13^{ème} mois de la même valeur + double pécule de vacances (92% du salaire mensuel brut de l'année précédente -> diminue si on n'a pas travaillé les 12 mois).

Coût annuel pour l'employeur :

C'est la somme du salaire annuel brut + les cotisations sociales patronales (25%) + les frais de secrétariat sociale (60€) + l'assurance-loi (235€) + les différents fonds sociaux.

Frais forfaitaires : C'est un remboursement de frais.

Sont exclues de la notion de rémunération, les sommes qui constituent un remboursement de frais dont la charge incombe à l'employeur.

Il ne peut s'agir exclusivement que des frais dont la charge incombe à l'employeur, c'est-à-dire ceux occasionnés par l'exécution du contrat de travail.

Les frais peu élevés qu'il est difficile de prouver à l'aide de justificatifs peuvent cependant être évalués de manière forfaitaire.

On y compte les déplacements, les absences de commodité des travailleurs itinérants, les frais de séjour, les voyages de services à l'étranger, les transports internationaux, les frais de bureau (télétravail), la connexion internet, les PC, les outils de travail, les achats de vêtements de travail, l'entretien des vêtements de travail, l'entretien et usure des vêtements du travailleur suite à un environnement très sale.

Le montant forfaitaire est fixé par la loi. **Ils s'ajoutent au revenu net (tout à la fin)**. Ils ne sont donc pas soumis à l'ONSS, au précompte professionnel mais par contre, il ne s'agit par d'une rémunération donc ils ne sont pas payés pour les congés, les maladies, le 13^{ème} mois, etc.

Chapitre 1 de la partie 2 : le développement logiciel -> l'ingénierie logicielle



Génie logiciel : (software engineering) étudie les méthodes de travail et les bonnes pratiques des ingénieurs qui développent des logiciels. Ces bonnes pratiques couvrent la conception, l'amélioration et l'installation des logiciels, afin de favoriser la production et la maintenance de composants logiciels de qualité. Il donne de manière abstraite une version de ce que l'on fait.

Pratique : une approche concrète et éprouvée qui permet de résoudre un ou plusieurs problèmes courants ou d'améliorer la façon de travailler lors d'un développement. Elle est acquise grâce à l'expérience. On finit par l'utiliser de manière consciente alors qu'avant, on le faisait inconsciemment.

Méthode de travail : elle fixe comment on fait les choses, quelles sont les responsabilités de chacun et les interactions entre les personnes. Elle est souvent vaste et complexe. Elle concerne plusieurs employés qui font chacun des activités différentes. Ces activités forment un tout. L'objectif est de fixer une démarche pour réaliser chaque activité dans les meilleures conditions.

Pourquoi décrire ses méthodes de travail ? Formalisation :

- 1) Dans une optique de transparence : il faut les comprendre, les partager et les communiquer dans l'équipe / entre entreprises.
Définir une méthode de travail permet de comprendre comment les activités s'articulent ensemble (organisation du travail). **Cela permet à chacun de connaître son rôle**, et que toutes les personnes ayant le même rôle travaillent de la même manière. Tout le monde doit utiliser le même git par exemple.
Définir une méthode de travail permet de la partager plus facilement et de faciliter l'apprentissage d'un nouveau membre de l'équipe.
Définir une méthode de travail permet de communiquer plus facilement entre plusieurs employés qui travaillent à des activités différentes mais dans un même but. Cela favorise l'utilisation d'un langage commun. Les langages sont différents d'une entreprise à l'autre (différentes littératures, abréviations, etc.). Chaque société développe un langage propre à tous ceux qui la compose.
- 2) Dans une optique de pérennité : parce que le temps passe, les équipes changent, les activités évoluent...
Permettre de suivre les processus et leurs changements au fil du temps.
Avoir l'assurance de ne pas perdre les connaissances lorsque quelqu'un quitte l'entreprise.
Que le savoir continue d'exister.
- 3) Dans une optique de satisfaction des clients : améliorer la réponse aux attentes du client. Ça se passe en 3 axes : besoins (specifications), budget (budget), délais (timeframe). Tout ça forme le processus.

Le génie logiciel touche au cycle de vie des logiciels.

Cycle de vie des gros logiciels : le nombre de personnes impliquées est grand et le temps de vie est long (au niveau du développement et de l'utilisation). Il est composé de différents processus, il a plusieurs versions, il y a beaucoup d'informations répartie entre différentes personnes et surtout, les spécifications et la validation sont essentielles.

Cycle de vie des petits logiciels : plus simple. On ne prend pas le temps d'écrire et de décrire les méthodes de travail et le cycle de vie.

Les activités du cycle de vie logiciel :



Les requirements du client vont amener à une étude de faisabilité qui enclenchera le cycle de vie. Le cycle de vie est composé des activités suivantes : spécifications, analyse formelle, conception, implémentation, tests unitaires, intégration et tests, livraison, installation et mise en production, utilisation-amélioration-maintenance.

Requirements :

Ce sont des demandes émanant du client (dans le cas d'un projet), du marché, de la concurrence, de la veille technologique, des partenaires, etc. (dans le cas d'un produit et éventuellement d'un projet).

Document remis par le client. Input du processus de développement logiciel.

Le souci est qu'ils sont souvent incomplets, parfois ambigus ou même parfois incorrects. Bêtement par exemple parce que l'utilisateur utilise un raccourci qu'on ne connaît pas. Il faut un certain type d'aménagement.

Tous les utilisateurs pourront effectuer des recherches

Spécifications :

C'est la traduction des demandes utilisateurs exprimées dans les « requirements ». Cette traduction est orientée vers le développeur logiciel. C'est l'élaboration du cahier des charges. L'objectif est de définir ce que le logiciel doit faire.

Analyse énoncé, définitions objectifs de l'application à développer, réflexion sur les IHMS, analyse par les données, etc.

L'output des spécifications sont le diagramme de use cases, la liste des objectifs, les prototypes d'écran, la définition des données → base de données, on a besoin de connaître la structure de données. L'analyse initiale est 80% des spécifications. Si c'est un client, c'est également le cahier des charges à communiquer au client sur lequel le client va s'engager. Si c'est un produit, c'est la liste des demandes.

Analyse formelle :

Si le client a accepté le cahier des charges, on va procéder à l'analyse formelle. La question est : que va faire le système ? On va comprendre clairement ce qu'est le système et tous ses concepts sous-jacents. On va vérifier que cela correspond à ce que demande le client.

L'output est une description de chaque fonctionnalité.

Conception :

Elle modalise et spécifie le comment. C'est l'architecture. On va organiser un système logiciel.

A la vue d'une analyse, il s'agit des diagrammes de classe – OO – responsabilités, collaboration, communication. C'est l'organisation interne (découper en composants). C'est l'intégration des composants externes : comment appeler leurs API ? Comment isoler ces appels ? Il faut faire attention à ce que l'API que l'on fournit pour nos clients puissent intégrer leur propre code (ex : validation d'un champ). Conception de l'API que l'on fournit pour être appelé en tant que composant d'un autre logiciel.

Implémentation :

C'est la construction du logiciel au niveau à la fois du codage, mais aussi l'intégration des composants externes. Il y a un sous-groupe pour chaque composant.

L'output est le code-source et la documentation (justification des choix, code commenté).

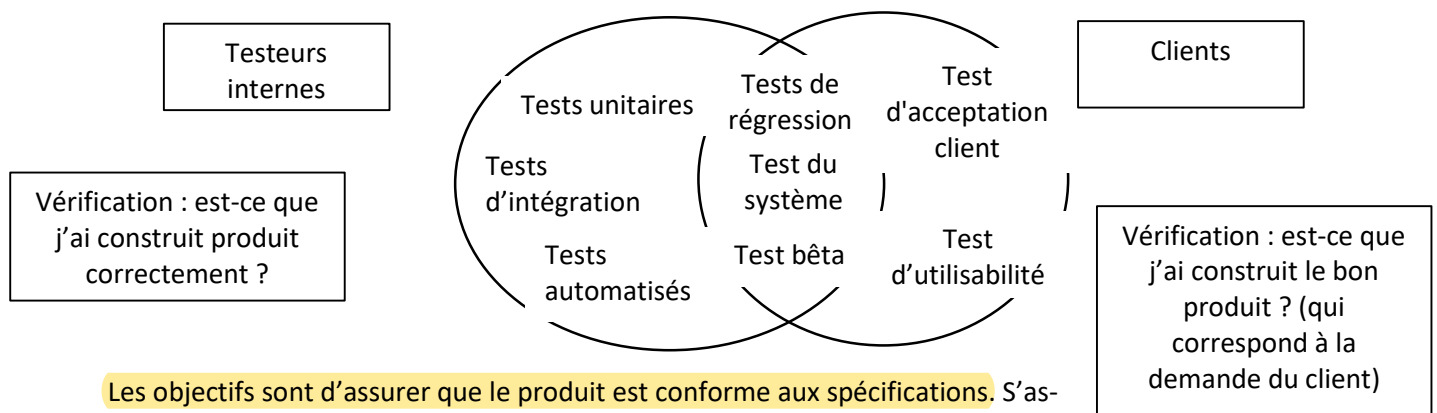
Tests unitaires :

C'est le test de blocs de code (et pas d'une fonctionnalité). On l'appelle aussi block box. Ce sont des tests écrits par les développeurs eux-mêmes pour tester leurs classes ou leur code. Ces tests sont effectués par des machines, donc par besoin qu'un utilisateur vienne. Ces tests font souvent partie de l'implémentation.

L'output est les tests eux-mêmes. C'est l'exécution des tests et l'enregistrement des résultats de ceux-ci.

Intégrations et tests :

Ça représente le double procédé de **vérification** et **validation** d'un logiciel.



Les objectifs sont d'assurer que le produit est conforme aux spécifications. S'assurer que le produit fait ce que l'on attend. Détecter les erreurs et les bugs. Et répondre à la question : est-on prêt pour l'étape suivante ?

Vérification : une batterie de tests →

- 1) Tests d'intégration :
Vérifie que les composants s'intègrent bien ensemble et que le produit est compatible avec l'environnement logiciel et matériel prévu chez le client.
 - 2) Tests fonctionnels :
Vérifie que le produit répond à l'analyse formelle (ou fonctionnelle).
 - 3) Tests système :
 - Tests de performance (temps de réponse à une requête)
 - Tests en volume (beaucoup de connectés ou beaucoup dans le DB en même temps)
 - Tests de stress (exagération de la demande : tout le monde clique sur « validez » en même temps)
 - Tests de fiabilité (résistances aux pannes : coupure réseau, etc.)
 - Tests de sécurité
 - Tests d'utilisation en « vitesse de croisière » (quand tout va bien)
- L'output ce sont les tests, l'exécution de ceux-ci et la documentation de l'exécution.

Validation :

Ou recette. Ce sont les tests de validation ou d'acceptation. Ils sont formalisés par le client et leur succès assure l'acceptation du logiciel par le client. L'output est l'exécution des tests par le client, le rapport et la signature pour acceptation du logiciel.

Livraison, installation et mise en production :

C'est la mise à disposition chez le client. L'output, ce sont le logiciel mis en production chez le client et un document signé par celui-ci attestant la mise en production. Ces outputs prouvent que le travail a été fait, accepté et confirmé.

Maintenance :

Ce sont les changements apportés au système après sa mise en production.

- 1) Maintenance corrective : correction de bugs ou de défaillance
- 2) Maintenance adaptative : adaptation de la solution à de nouvelles contraintes techniques (plus de Windows 7)
- 3) Maintenance évolutive : modifications du logiciel entraînées par des changements ou ajouts dans les besoins

Le développement logiciel :


Les activités s'organisent dans le développement logiciel de la manière suivante :

- le processus management s'occupe de planifier le travail, de planifier les livraisons, d'allouer les ressources, de gérer le budget et les coûts, de surveiller l'avancement des travaux et de gérer les risques
- le processus de développement s'occupe de créer le logiciel selon les demandes du clients (requirements)

La question est de savoir maintenant, dans quel ordre se passe toutes les activités vues plus haut.

Cycles de vie classiques :

Cascade :

Dans l'ordre  l'on a vu, chaque étape doit être finie avant qu'une autre ne commence. Chaque étape est réservée à une équipe. Chaque équipe reçoit les documents des étapes précédentes. En théorie c'est parfait mais en pratique, on n'a jamais le moyen de retourner au début s'il y a une erreur. C'est une très mauvaise gestion du risque. On n'a pas de gestion de changement qui soit possible. Ce n'est pas optimal car les requirements du client peuvent changer ou être mal compris.

- Avantages : une documentation est produite entre chaque phase (analyse formelle, choix de conception, documents de tests, description du projet/produit à livrer). C'est bien supporté par les outils de planification. C'est efficace quand les développements sont complexes.
- Inconvénients : le modèle n'est pas réaliste. Les spécifications sont figées beaucoup trop tôt dans le processus. Les spécifications sont validées beaucoup trop tard. Il n'y a ni gestion de changement, ni gestion de risques. Il n'y a pas de mise à jour de l'analyse en cours de développement. Et il n'y a pas de tests en cours de développement.

Une amélioration possible peut être de permettre les retours. Mais le souci est toujours que chaque étape doit être réalisée en une fois et ne peut commencer que lorsque la précédente est finie. La validation est toujours trop tardive.

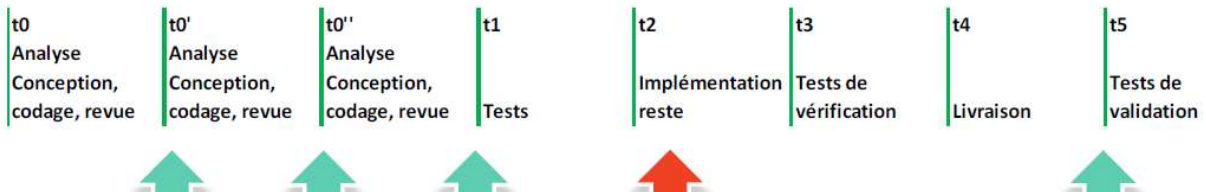
Dans le temps : chaque étape fait $t+1$. Les points de synchronisation sont l'attente qu'une étape soit terminée. Les outputs sont les inputs de l'étape suivante.

Rapid Application Development (RAD) :

Le but ici est de mettre la gestion du risque au centre. On évite le risque que le client n'accepte pas. On teste via des prototypes les parties difficiles du système et beaucoup de fonctionnalités. La boucle est extrêmement rapide.

- Avantages : livraison à temps, validation dès le début du développement
- Inconvénients : manque de documentation (car ça a été trop vite), système avec des structures très pauvres et les changements sont tout aussi difficiles

Dans le temps :



Les flèches bleues sont l'interaction avec le client. La rouge, on se demande si l'analyse est suffisante et si la conception est suffisante.

Modèle en Y :

C'est la montée en puissance de l'architecture. On réduit le risque technologique en réintroduisant les bonnes pratiques.

Dans le temps :



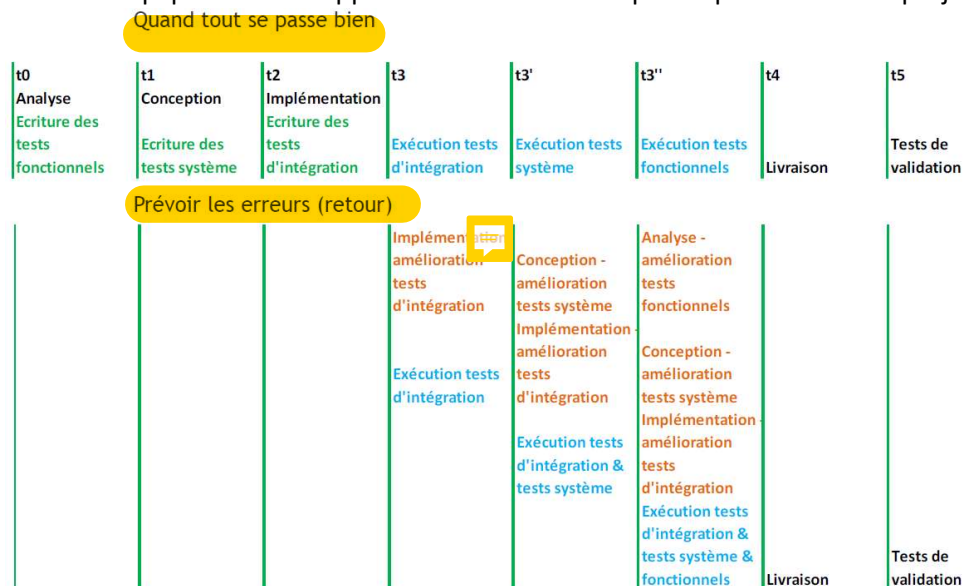
L'interaction avec le client arrive tard avec la validation.

Modèle en V :

Le plus courant. Il essaie de prendre les points positifs de chaque modèle. Le but est d'écrire les tests au moment où on confectionne la partie visée. Comme ça, s'ils ne passent pas, on reconfectionne. Dans ce schéma, les tests ont la même importance que tout le reste. L'exécution est fort tardive mais au moins, on ne les a pas écrits au moment des tests, le code ne sera donc pas influencé.

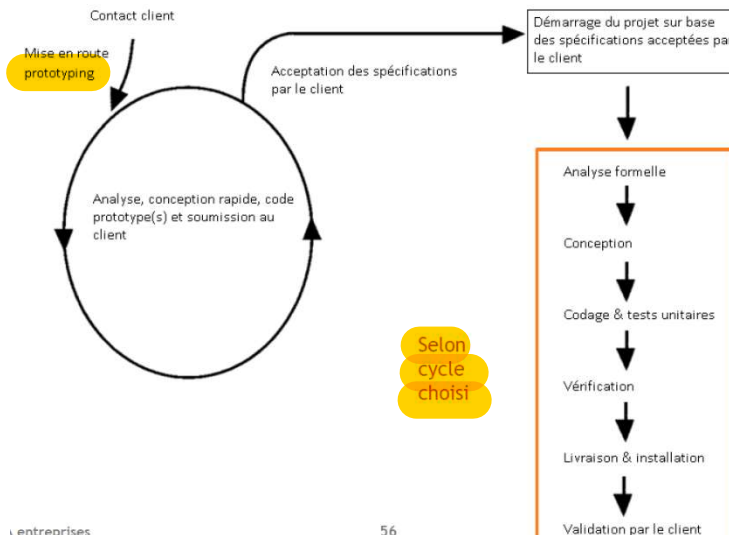
- Avantages : toute étape descendante est accompagnée de l'écriture des tests qui permettront d'assurer qu'un composant correspond à sa description. Cette méthode de travail rend explicite la préparation des dernières phases (vérification, validation) par les premières (construction du logiciel). Toute propriété du logiciel DOIT être vérifiable objectivement après la réalisation. La planification est aisée. L'organisation est facilitée entre les différentes équipes. Les outputs sont clairement définis. Le client est associé dès le début du projet pour écrire les tests de validation.
- Inconvénients : le client n'intervient pas en cours de projet. Le client découvre souvent trop tard les erreurs. Les équipes de développement doivent être expertes pour évaluer le projet.

Dans le temps :



Modèle exploratoire :

L'idée est de combiner de 1, **le prototyping** (partie exploratoire) : les spécifications sont obtenues à partir de la maquette. L'objectif est d'avoir des spécifications adéquates que le client accepte le plus rapidement possible. Il faut aussi la meilleure adéquation entre les spécifications et les besoins réels. Et de 2, une **autre méthode de développement** (cascade, Y, V) : On jette le code écrit précédemment et on utilise l'expérience du prototyping pour valider les spécifications et on démarre ensuite le SDLC selon la méthode choisie.



- Avantages : les spécifications sont bien comprises par toutes les parties (client, chef de projet, développeurs, testeurs, etc.).

Le choix du cycle de développement après acceptation des spécifications conditionne les autres avantages et inconvénients.

Modèle en spirale :

Par exemple, le projet est découpé en 3 cycles. Chaque cycle contient les 6 phases. Après chaque cycle, il y a une livraison au client.

- Avantages : on ne doit pas avoir **analysé l'entièreté de la demande du client dans les détails**, on le fera pour une partie du développement à chaque cycle. On peut profiter de l'expérience du développement des cycles précédents pour évaluer les alternatives offertes pour le développement du nouveau cycle. **Chaque cycle se termine par une version livrable**. Le client peut valider le logiciel à chaque cycle. Le temps de développement d'un cycle est plus court et les risques mieux gérés à l'intérieur du cycle.
- Inconvénients : la gestion de la découpe des spécifications est complexe car il faut tenir **compte des interactions entre les fonctionnalités**, des difficultés rencontrées pendant les cycles précédents, des objectifs et contraintes du cycle courant, des alternatives de réalisation des objectifs du cycle courant → la décision du développement à faire pendant un cycle doit tenir compte du risque. La planification est délicate. La conception peut être revue à cause des nouvelles spécifications.

Chaque cycle de vie de la spirale peut être développé selon n'importe quel procédé.

Dans le temps : **il y a une validation par le client à chaque fin de cycle.**

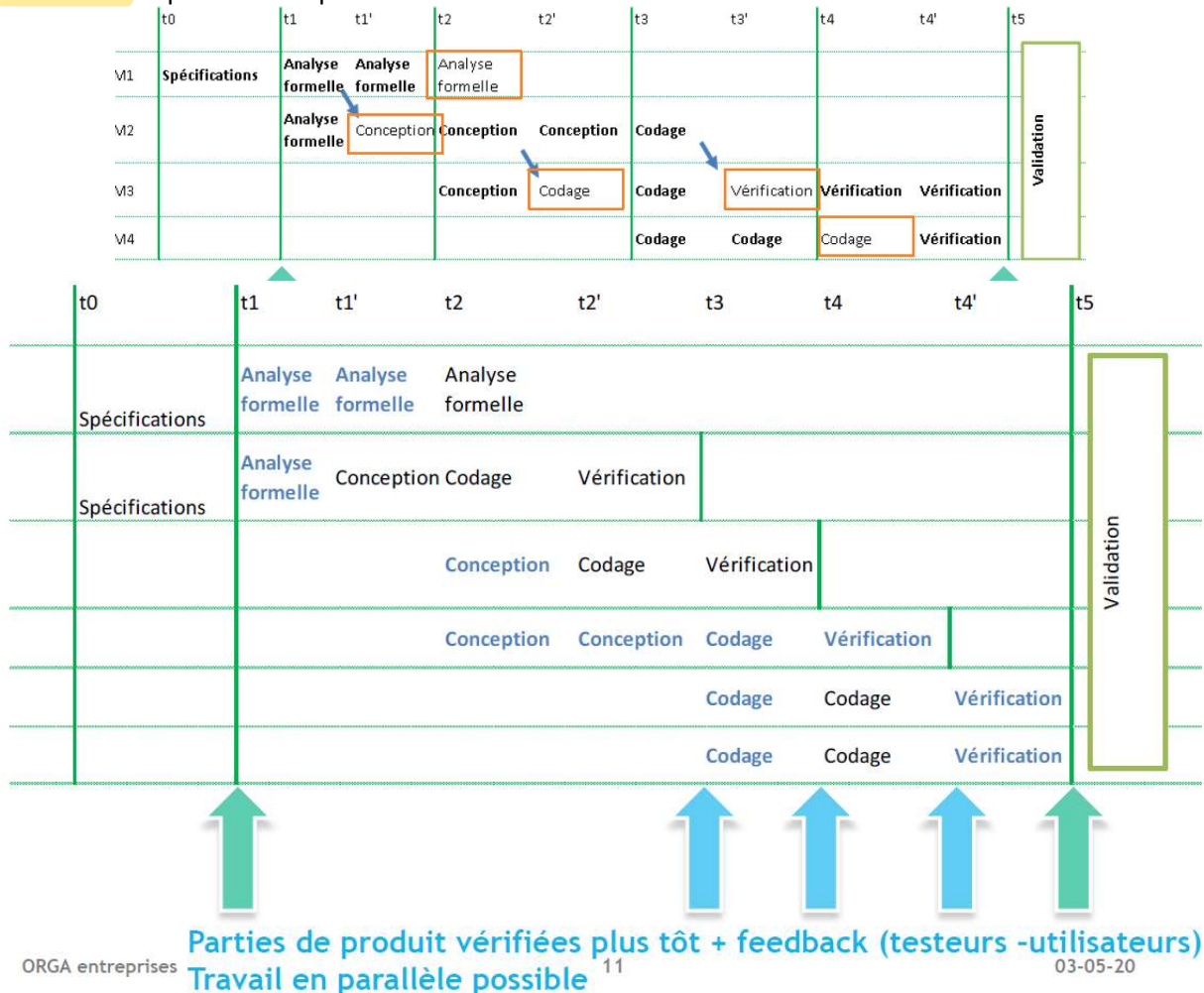
Cycle 1						Cycle 2						Cycle 3						Cycle 4					
t0	t1	t2	t3	t4	t5	t0	t1	t2	t3	t4	t5	t0	t1	t2	t3	t4	t5	t0	t1	t2	t3	t4	t5
Spécifications	Analyse formelle	Conception	Codage et tests unitaires	Tests de vérification	Tests de validation	Spécifications	Analyse formelle	Conception	Codage et tests unitaires	Tests de vérification	Tests de validation	Spécifications	Analyse formelle	Conception	Codage et tests unitaires	Tests de vérification	Tests de validation	Spécifications	Analyse formelle	Conception	Codage et tests unitaires	Tests de vérification	Tests de validation

Cycle de vie (2) :

Jusqu'à présent, théoriquement, on insiste sur la découpe en activités qui se suivent séquentiellement. Chaque étape est en t+1. On fait en une étape toutes les Use Case (Le cas d'utilisation correspond à un ensemble d'actions réalisées par le système en interaction avec les acteurs en vue d'une finalité. L'ensemble des cas d'utilisation permet ainsi de décrire les exigences fonctionnelles d'un système en adoptant le point de vue et le langage de l'utilisateur final), en une étape toute la structure, en une étape tout le développement. Les points de synchronisation sont l'attente qu'une étape soit terminée. Il y a seulement 2 contacts avec le client : un lors de la conception du cahier des charges, l'autre lors de la livraison du logiciel prêt.

Les retards pouvaient se gérer de 2 manières différentes : soit la date de remise était rallongée et toutes les étapes suivantes reculent d'une semaine, soit on engage un développeur qui vient renforcer l'équipe pour récupérer le retard de l'étape précédente dans celle-ci.

En pratique, c'est bien différent : il y a toujours 2 points de contact avec le client mais il y a plus d'élasticité et peu de réels points d'attente →



Les modèles à incréments :

- **Processus unifié** : unifié car dans les années 90, il y avait une 50aine de méthode orientées objet. On a donc recherché un langage commun : UML, qui est un ensemble d'outils normalisés, mais on a besoin d'une méthode. Il est composé d'une méthode, d'une couverture complète du SDLC pour les développements orientés objet ainsi que d'un lien avec l'UML. Méthode :
 - Il est piloté par les cas d'utilisation : le but est de comprendre les besoins des utilisateurs. Un cas d'utilisation est une fonctionnalité qui produit un résultat pour l'utilisateur.

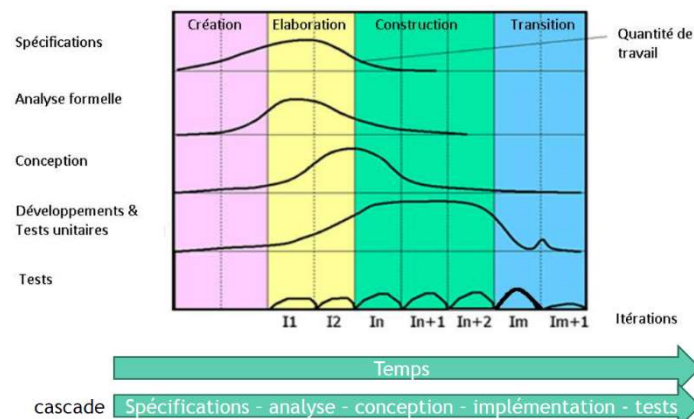
- Il est centré sur l'architecture logicielle : l'architecte dessine une image complète d'un bâtiment avant le début de la construction. C'est donc une image complète du système avant son implémentation.
- Il est à base de composants
- C'est une méthode de développement du logiciel itérative et incrémentale : l'idée de base est de 1) développer un système au travers de cycles répétés (itération) et en petites avancées (incrément). 2) Chaque itération peut reprendre plusieurs activités (activités qui vont des spécifications jusqu'à la vérification/validation). 3) Chaque incrément va ajouter de nouvelles fonctionnalités, c'est une construction morceau par morceau.

L'avantage majeur est qu'on peut tirer avantage de ce que l'on a appris durant l'itération précédente et que ça **réduit les risques** ? → on prend en charge très tôt dans le processus de développement les risques importants. On définit une architecture qui guidera le développement logiciel. On fournit une infrastructure préfabriquée (framework) pour prendre en compte non seulement les exigences de base mais aussi les changements futurs. On développe progressivement le système, de façon incrémentale.

Le processus unifié se déroule en phases :

- 1) La création (inception) : la vision du projet est encore approximative. On y élaborera surtout les cas d'utilisations.
Le **but** est de développer la vision du projet, de définir la portée du projet, de réduire les risques majeurs et de s'assurer de la viabilité commerciale. Tout est en une seule phase : pas d'itération.
Les **activités** sont de comprendre les besoins du client, de spécifier.
- 2) L'élaboration : la vision y est plus élaborée. Le noyau du projet sera implémenté, les risques élevés résolus. La plupart des besoins seront identifiés.
Le **but** est de développer l'architecture de référence, d'avoir compris l'essentiel des besoins, de réduire les risques élevés (risques de moindre gravité qu'en phase de création). Il peut y avoir plusieurs itérations.
Les **activités** sont surtout d'analyser et concevoir. Mais on fait toujours les 2 d'avant et on commence tout doucement à développer.
- 3) La construction : implémentation des éléments de risque et complexité plus faibles. Préparation du déploiement.
Le **but** est de développer le système, de réduire les risques, de vérifier l'utilisabilité du produit. Il peut y avoir plusieurs itérations.
Les **activités** sont surtout de développer et faire des tests, mais on fait toujours les 2 avant.
- 4) La transition : β -tests et déploiement.
Le **but** est de s'assurer que le produit est livrable, de déployer, de former les utilisateurs, de mettre en production. Il peut y avoir plusieurs itérations.
Les **activités** sont surtout de déployer et livrer. Tout le reste est presque effacé.

Son cycle de vie :



Pour conclure, le modèle à incréments supprime les points de synchronisation et les attentes entre étapes.

Approche Agile :

Les méthodes agiles sont un type de développement itératif, incrémental et adaptatif.

Agilité : capacité à favoriser le changement et y répondre en vue de s'adapter au mieux à un environnement turbulent.

Manifeste pour les développements Agile : Nous, par la pratique et en aidant les autres à le faire, reconnaissons la valeur des seconds éléments, mais privilégions les premiers.

- **Les individus et leurs interactions plutôt que les processus et les outils :**
Equipe de développeurs soudés et qui communiquent. Equipe de développeurs de différents niveaux qui communiquent bien entre eux plutôt qu'une équipe composée d'experts fonctionnant chacun de manière isolée. → COMMUNICATION
- **Les logiciels opérationnels plutôt qu'une documentation exhaustive :**
D'abord un logiciel fonctionnel. Importance du logiciel documenté (code) mais pas les documents fournis relatifs au projet : logiciel testé et documenté pour pouvoir le maintenir. Il faut minimiser les documents projets tels que les rapports d'avancement, notes explicatives, planning revu et maintenu à jour en permanence, flux pour signature, ...
- **La collaboration avec les clients plutôt que la négociation contractuelle :**
Le client doit être impliqué dans le développement. On ne peut négliger les demandes du client. Le client doit collaborer avec l'équipe et fournir un feed-back continu sur le logiciel. → LOGICIEL ADAPTÉ AUX ATTENTES DU CLIENT
- **L'adoption aux changements plutôt que le suivi d'un plan :**
Réagir aux demandes de changement. Planification flexible : plan de développement au début du projet, revu et remanié à chaque nouvelle itération : permettre l'évolution de la demande du client tout au long du projet et prendre en compte les demandes d'évolution provoquées par les premières releases (communiqués) du logiciel.

Principe Agile :

Le but est de **satisfaire le client**, c'est la plus haute priorité. Pour ça, on livre **rapidement** des fonctionnalités à **grande valeur ajoutée** (moto).

On **accueille les changements** de besoins, même tard dans le projet, pour donner un avantage compétitif au client : conception orientée évolution.

On **livre fréquemment** un logiciel opérationnel avec des cycles de quelques semaines à quelques mois. Pour ça, on met en production rapide une version minimale du logiciel et ensuite, des nouvelles livraisons incrémentales.

On **collabore quotidiennement** entre utilisateurs (ou leurs représentants) et développeurs.

On réalise les projets avec des personnes motivées, en fournissant l'environnement et le soutien dont elles ont besoin et en leur faisant confiance.

On communique par des **conversations en face à face** !

On **mesure l'avancement** par un logiciel opérationnel.

On garde un **rythme de développement soutenable** : maintenir indéfiniment un rythme constant.

On recherche **l'excellence technique et une bonne conception**.

On recherche la **simplicité**, c'est-à-dire de minimiser la quantité de travail inutile.

On laisse l'équipe **s'auto-organiser**, on n'impose pas de processus.

On laisse l'équipe réfléchir aux moyens de devenir plus efficace et adapter son comportement.

Agile : adaptatif :

- 1) Planification :

Elle est **itérative**. C'est-à-dire que l'on va répartir de manière tout à fait mathématique chaque UC et chaque T. Le planning initial n'est pas détaillé, on vérifie simplement et rapidement que l'on peut tout développer.

A la fin de chaque itération, on regarde ce que l'on a réalisé et on adapte le planning. Par exemple, si l'on n'a pas réalisé la UC de la semaine 2, on la reporte en semaine 3 et la UC initiale de la semaine 3 est reportée aussi. Mais le planning est mis à jour pour seulement une itération (itération de 2 semaines) et avec le client.

2) Fonctionnel :

Par adaptation systématique du produit aux changements de besoin. La demande de changement est bienvenue mais de un, le changement n'est pas permanent, ensuite il ne peut y avoir d'interruptions intempestives du développeur qui travaille et l'engagement pour l'itération en cours n'est pas modifié. On passe par **une gestion des priorités** et la demande de changement peut être différée.

3) Technique :

Par remaniement régulier du code : **refactoring** → c'est refondre le code source pour en améliorer la qualité et changer le design sans changer les fonctionnalités. Le refactoring est fort encouragé dans les méthodes Agile. C'est un outil de changement.

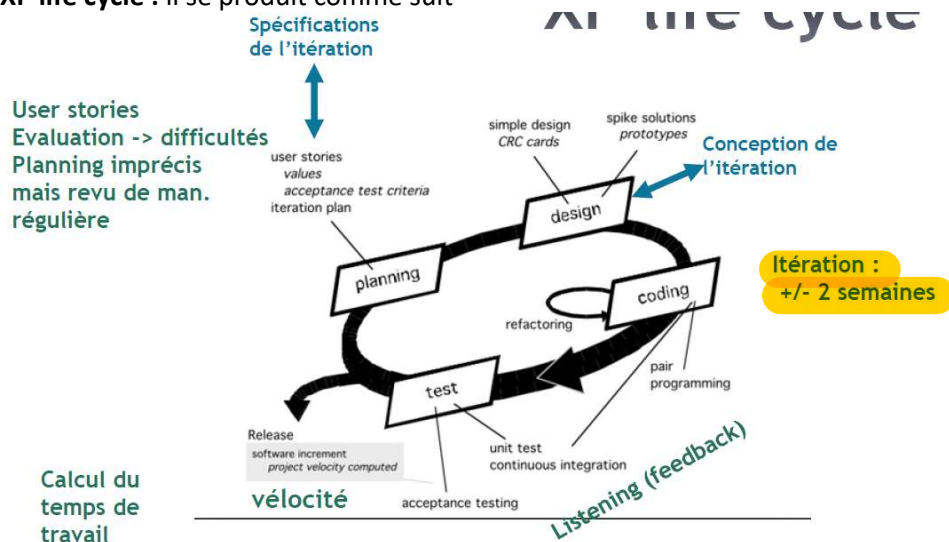
Deux méthodes agiles :

1) eXtreme Programming :

Pratiques de programmation : La conception est simple. On fait du **refactoring** (code en permanence remanié pour rester simple et compréhensible). On fait **des tests de non-régression**. Ces batteries de tests permettent de faire face aux modifications permanentes. On fait des tests de vérification et d'acceptation. Ces batteries de tests sont automatisées.

Pratique de collaboration : La programmation se fait en **binôme**, ce qui veut dire qu'il y a une revue du code en permanence et qu'il y a un apprentissage et un partage des connaissances. La responsabilité du code est collective, car chaque développeur est responsable d'une partie du code mais en même temps, ils sont tous responsables du code puisque chacun y a accès. L'intégration du code est continue car on intègre des nouveaux développements chaque jour.

XP life cycle : il se produit comme suit



User stories : Il s'agit d'un format associé aux méthodes agiles. C'est une description brève d'une fonctionnalité telle que vue par l'utilisateur. C'est un format écrit court, laissant de la

place à la discussion orale. Ça permet une émergence rapide dans des ateliers collaboratifs. C'est d'une grande simplicité et donc d'une grande lisibilité. Histoire implémentée en une seule itération (découpe si besoin). Une user story est utilisée en tant que spécification mais aussi pour l'estimation du temps de développement et la planification. Elle possède un ID, un nom (description brève), une histoire (en tant que, je veux, afin de), une validation (vérification que tous les attributs du client sont là, l'obligation des données et les liaisons, en gros un DSD) et un poids (effort estimé au moment de la création de l'US, parfois estimation ou laissé vide et sera apprécié au moment du planning de l'itération).



Vélocité : on constate au début de l'itération que chaque US a un poids. On planifie un nombre d'US équivalent à notre vélocité. A la fin de l'itération, la somme des poids des us EFFECTIVEMENT RÉALISÉES est égale à la vélocité de l'itération (dont on a besoin au début). La vélocité est une constatation à postériori.

Début itération

- User Story A : 2 points
- US B : 3 points
- US C : 3 points

Fin d'itération

- US A : terminée
- US B : non terminée
- US C : terminée

→ Vélocité = 5 points.

On prévoit donc l'itération suivante : 5 points.

//slide 29

2) Scrum :

C'est une méthode agile qui apporte plus de valeur aux clients et aux utilisateurs en maximisant la valeur ajoutée → en réalisant d'abord les fonctions à plus haute valeur ajoutée et en changeant les priorités et même les fonctions. Cette méthode apporte une plus grande satisfaction dans le travail car c'est une équipe auto-organisée.

Adaptatif : Scrum fournit un cadre pour le développement d'un produit complexe. L'idée est qu'il est impossible de définir tous dès le début car les spécifications peuvent changer, des outils ou technologies inconnus entreront en jeu, etc. Pour s'adapter aux changements, les travaux à faire sont ajustés à la fin de chaque itération.

Cadre : Il est très léger → il y a 3 rôles, c'est sous forme d'itérations, il y a des réunions au début et à la fin de chaque itération, il y a une mêlée quotidienne, il y a un backlog de produit et il y a peu de production documentaire.

Rôle 1 -> product owner : C'est le directeur du produit. C'est le représentant du client et des utilisateurs dans l'équipe. C'est le représentant ayant la vision du produit et l'autorité pour donner les priorités aux requirements du client. C'est le responsable du Product Backlog. C'est le responsable du résultat auprès des autres parties. Il faut le canal de communication avec les parties prenantes.

Rôle 2 -> scrum master : C'est le gestionnaire. Il facilite l'application de Scrum dans l'équipe. Il assure que la méthodologie soit respectée et provoque les changements organisationnels nécessaire à cela. Il supprime ce qui pourrait interrompre les équipes et les protège des interférences extérieures. Il assure l'amélioration des pratiques et de l'organisation du travail. C'est le coach de l'équipe.

Rôle 3 -> l'équipe de développement : C'est l'ensemble des individus participants aux activités de développement. Ce sont des individus ayant des compétences transversales. L'équipe est performante et auto-organisée. L'équipe est centrée sur les livraisons.

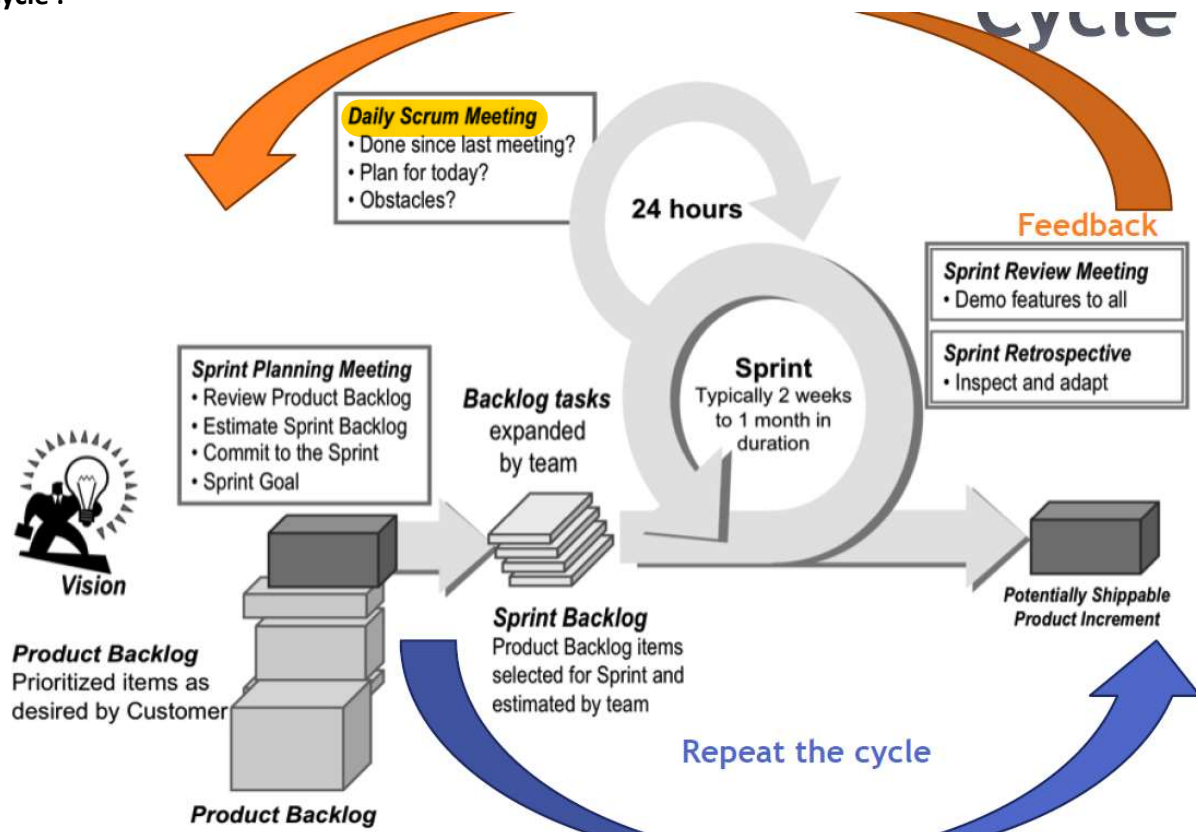
Quelques concepts :

- Sprint : unité de temps qui permet de rythmer les développements, correspondant à une itération. Ils sont de durée courte (max 1 mois), il n'y a pas de chevauchement,

le rythme est régulier (tous les sprints sont de même durée), il n'y a pas de changement de date de fin si le développement n'est pas fini et ça fixe la quantité de travail à produire en fonction de la durée et de la taille de l'équipe. Le budget est fixe (si la taille de l'équipe est stable mais ceci est recommandé).

- Product backlog : liste, **ordonnée par priorité**, des fonctionnalités pour le produit, définie par le directeur de produit. Les nouvelles demandes ou corrections de bugs sont ajoutées dans le product backlog.
- Sprint backlog : liste des fonctionnalités qui sont développées dans le sprint.
- Version : plusieurs sprints peuvent être nécessaires pour développer une version du produit.

Cycle :



Réunion de début de sprint (sprint planning) :

La planification du sprint se fait par la direction du ScrumMaster. On fait la définition de l'objectif du sprint (Spring Goal). On analyse du haut de la liste du ProductBacklog. On définit le sprint backlog en fonction de la capacité de l'équipe et de la priorité des tâches. Pour cela, il va falloir estimer **le poids des user stories en fonction de la complexité**, la longueur et les risques de complication, la division éventuelle, consensus sur le poids, comparaison avec vélocité (moyenne des 3 derniers sprints par exemple).

Mêlée journalière :

L'objectif est de **savoir ce que chacun fait**, se synchroniser, mesurer l'avancement et s'entraider. Tout d'abord, on fait un rapide tour d'avancement, où tout le monde est debout. C'est une réunion une fois par jour à heure fixe. On met en commun les apports de chacun et on partage les difficultés rencontrées. Ensuite, on répond à 3 questions : Qu'as-tu terminé depuis la précédente réunion ? Que penses-tu pouvoir terminer d'ici la prochaine réunion ? Quels obstacles rencontres-tu en ce moment ? Attention aux risques de ne pas respecter le

sprint backlog. Pour finir, il y a des ajustements qui peuvent être fait en fonction de ce que le Scrum Master détermine comme avancement par rapports aux engagements.

Réunion de revue :

C'est la démonstration des nouvelles fonctionnalités (en interne avec l'équipe). A ce moment-là, le directeur de produit fait un feedback. On redéfinit ensuite les priorités du productBacklog et on identifie les user stories à traiter lors du prochain Sprint.

Rétrospective :

C'est une prise de recul sur l'itération qui s'est terminée. On identifie des améliorations potentielles (processus, méthode de travail). On détermine le « comment » réaliser ces améliorations → ce que l'on aimerait mettre en place, ce que l'on souhaite arrêter, ce que l'on souhaite continuer. Cette rétrospective permet donc d'améliorer le fonctionnement de l'équipe et d'augmenter la satisfaction de ses membres.

Cycle et Sprint :

A l'intérieur d'un sprint, on va développer un produit partiel, c'est-à-dire spécifier, analyser, concevoir, développer, tester, documenter et intégrer pour chaque US. Pendant le cycle des sprints, les spécifications sont continues, c'est-à-dire que le client précisera ses demandes au moment où elles sont développées. La conception est elle aussi continue, c'est-à-dire que l'architecture va évoluer pendant la vie du projet. Les tests sont présents dès les premiers sprints.

Finish :

Les sprints se répètent jusqu'au moment où il y a eu assez de fonctionnalités développées dans le Product Backlog ou quand le budget est épuisé ou quand la date limite du projet est atteinte. Ça n'est pas grave car les fonctionnalités à plus haute valeur ajoutée ont été développées.

Périodes :

Il y a 3 périodes en Scrum :

- Avant le premier sprint, après discussions avec le client et la remise de prix, on fait la définition du backlog initial, la définition de la vision, la constitution de l'équipe, la mise en place de l'environnement, la maîtrise de l'architecture et la première planification générale.
- Les sprints : période centrale.
- Une période de clôture composée des tests d'acceptation sur tout le produit/projet/release, le déploiement de l'environnement de production, l'écriture des manuels et le fait de donner la formation.

Exemples :

Le scrum board contient la répartition des tâches. Elles sont souvent identifiées par des couleurs.

PU VS Scrum : PU est une définition complète des objectifs du projet. Il a 4 grandes phases. Le planning c'est seulement la date de fin du projet qui est définie. Les outputs sont très bien définis.

Pour le Scrum, chaque itération couvre le cycle complet pour les US. C'est le product owner qui détermine quand le projet est fini. Les outputs sont le code fonctionnel et documenté ainsi que les backlog à jour.

EXAMEN → comparer les méthodes pour pouvoir choisir une méthode.

Choix d'une méthode d'ingénierie logicielle :

Cascade, V ou Y :

Ces 3 méthodes ont une approche très contrôlée du développement de par une planification précise, des méthodes d'analyse et de conception, de la documentation. Elles sont faciles à comprendre. Les points négatifs sont qu'elles ont une attente entre 2 étapes, il y a un manque d'adaptabilité et des problèmes vus à la validation (client).

On remarque que ces 3 méthodes passent beaucoup de temps sur la façon de développer un système avant le développement lui-même.

Itératif (modèle à incréments) :

Les avantages sont :

- Une première version est fournie rapidement
- Les risques d'échec sont diminués car on découvre les problèmes assez tôt, les parties importantes sont développées en premier, les coûts des risques sont limités à une itération, la gestion de la complexité et le rythme de développement sont soutenus grâce à des objectifs clairs et à court terme, et pour finir, les avancées sont évaluées au fur et à mesure de l'implémentation.
- Les progrès sont visibles rapidement
- Les tests et intégration se font de manière « continue »


Les désavantages sont :

- La définition de l'itération est délicate, elle demande du temps et est risquée
- Le modèle est lourd à mettre en œuvre

C'est donc inadéquat pour les petits projets.

Agile :

Les avantages sont :

- Méthodes focalisées sur le développement
- Méthodes basées sur une approche itérative
- La livraison est rapide et on a un feedback des clients 
- C'est un **développement d'application dont les exigences changent**

Les désavantages sont :

- Il n'y a souvent aucune documentation
- Il y a des difficultés à mettre en œuvre l'affectation des priorités, la simplicité dans les changements additionnels, l'implication intense des développeurs
- Une frontière ténue peut advenir entre l'application de la méthode et le « n'importe quoi » quand on travaille sans méthode.

Scrum contesté :

Les obstacles sont que l'équipe travaille dans différents lieux, que les membres de l'équipe viennent d'organisations différentes et que les membres de l'équipe n'ont pas le pouvoir de prendre les décisions.

//slide 8 à revoir

Quel modèle choisir ?

Il n'existe pas un modèle idéal à appliquer en toute situation. Il faut faire un compromis en fonction du contexte. Adopter un cycle de vie est déjà une preuve de maturité pour l'entreprise.

Comparatif

	Discipline	Agile
Taille (de l'équipe)	<ul style="list-style-type: none"> Nécessaire si grosses équipes Difficile à adapter aux petits projets 	<ul style="list-style-type: none"> Petites équipes Forte dépendance au mode de connaissance tacite qui limite la taille DANGER
Criticité (pertes en cas de défaut)	<ul style="list-style-type: none"> Idéal pour gérer le développement de produits critiques Convient moins aux produits peu critiques 	<ul style="list-style-type: none"> Non testé sur des systèmes critiques Difficultés potentielles dues au simple design et au manque de documentation
Dynamisme (% de changement d'exigences/mois)	<ul style="list-style-type: none"> Plans détaillés et design précis sont bien adaptés aux environnements stables Mais source de travail importante quand l'environnement est changeant 	<ul style="list-style-type: none"> Simple design & refactoring continu sont bien adaptés aux environnements forts changeants Mais peut être source de travail supplémentaire dans un environnement stable
Personnel (expérience)	<ul style="list-style-type: none"> Demande du personnel expérimenté durant la définition du projet Utilisation de juniors par la suite 	<ul style="list-style-type: none"> Demande la présence d'une masse critique de personne expérimentées Risqué avec des juniors
Culture	<ul style="list-style-type: none"> Organisations où les rôles sont clairement définis dans des procédures reconnues 	<ul style="list-style-type: none"> Adapté aux organisations laissant une grande marge de liberté aux employés

Tests fonctionnels :

Les tests sont une activité dans laquelle un système ou un composant est exécuté **dans des conditions spécifiées**, les résultats sont **observés ou enregistrés**, et **une évaluation est faite** pour certains aspects du système ou du composant.

➔ **Pour pouvoir comparer les résultats obtenus à ceux prévus.**

L'objectif des tests est de vérifier la conformité d'un logiciel avec ses spécifications. Le but est aussi de trouver des anomalies qui n'ont pas été détectées. Par exemple introduire un numéro de client, qu'il soit obligatoire, avec une lettre et 4 chiffres, cette lettre correspondant à la province Belge dans laquelle le client vit. Le test vérifiera qu'il est possible d'entrer un numéro de client, et qu'il est impossible d'enregistrer un client français avec seulement les 4 chiffres.

Les tests servent également à vérifier que le logiciel réagit de manière conforme :

- Les tests de vérification visent ainsi à vérifier que ce système réagit de la façon prévue par ses développeurs.
- Les tests de validation visent ainsi à vérifier que ce système est conforme aux besoins du client.

Les tests de vérification sont composés des tests unitaires, des tests d'intégration, des **tests fonctionnels**, des **tests systèmes**.

Les tests de validation sont formalisés par le client et sont composés des **tests fonctionnels** et des **tests système**.

Les tests fonctionnels :

Leur objectif est de vérifier la conformité d'un logiciel avec ses spécifications (cahier des charges et analyse formelle). Ils mettent en œuvre des scénarios qui permettent de tester les tâches-métiers (fonctions) pour lesquelles le logiciel a été développé. Le but est de valider l'exécution de la tâche-métier.

Ils mettent l'utilisateur en situation de travail. Ils sont souvent développés en technique de « boîte noire ».

Chaque test fonctionnel valide une partie du logiciel (journée entreprises) : soit une action isolée (validation du nom de l'entreprise), soit le scénario d'un cas d'utilisation (créer une entreprise), soit un enchaînement de cas (scénario d'usage)(indiquer le nom des personnes présentes lors de la confirmation de la participation de l'entreprise à 1 JE).

Couverture fonctionnelle :

Chaque fonctionnalité métier de l'application doit être vérifiée par au moins un **cas de test**. Par exemple, un algorithme permettant de calculer cela (partitionnement, classe d'équivalence, etc.).

Un cas de test est un « ensemble d'entrées, de conditions d'exécution et de résultats attendus, développé pour un objectif particulier, tel que le cheminement particulier d'un programme, ou la vérification de la conformité à une exigence spécifique.

Chaque fonctionnalité métier de l'application vérifiée par un cas de test et l'ensemble de toutes les combinaisons possibles d'emploi des fonctionnalités du logiciel (ordre de saisie des données, ordre d'apparition des erreurs, chemins, ...) sont la couverture fonctionnelle. On teste d'abord les chemins pris par un utilisateur (façon de travailler et ensemble des cas limites).

Approches :

Il y a l'approche manuelle centrée sur l'IHM. Elle est intuitive et permet de reproduire le comportement de l'utilisateur final. Elle permet de tester les cas d'utilisation d'une application (vue de l'utilisateur).

Il y a l'approche automatisée par programmation. Elle demande des capacités que n'ont souvent pas les clients finaux. Elle sera utilisée par des développeurs. Elle est souvent utilisée en Agile pour relancer les tests fonctionnels à chaque build ou à chaque itération.

Faire évoluer les tests :

Les tests doivent évoluer en fonction des modifications apportées au logiciel. Il faut garder les données qui ont permis de reproduire un bug et le scénario de test (ou CR). Il faut les intégrer dans le scénario global de tests de l'application. Ceci permettra de tester la non-réapparition du bug (non-régression des développements futurs).

Rédiger les tests :

Les tests peuvent être exécutés de nombreuses fois pendant la durée de vie du logiciel. Il faut prévoir que le document puisse être rempli, commenté, daté et signé à chaque exécution. Le document est une représentation des tests à exécuter, son remplissage se passe lors de l'exécution des tests.

Il y aura plusieurs documents de tests.

Document de tests :

Description de l'environnement, référence au(x) cas d'utilisation, liste des prérequis, description des données en input (déjà présentes avant l'exécution du cas), décrire le scénario du test, enfin description des données d'output.

Scénario de test :

Il comprend plusieurs étapes : cas de test le plus général dans lequel tout se passe bien, toutes les autres combinaisons possibles dans lequel tout se passe bien, cas d'erreurs.

Cas général, exemple : créer un nouvel utilisateur → utilisateur créé dans la DB.

Cas alternatif, exemple : créer un nouvel utilisateur mais ne pas remplir les champs dans l'ordre logique → utilisateur créé dans la DB.

Cas d'erreurs :

Il y a des tests des erreurs isolées, comme par exemple une saisie de données erronées ou une absence de valeur.

Il y a des tests de cas extrême : fourchette avec un minimum plus grand que le max, valeurs autorisées (valeur minimale, valeur max + 1, combinaisons inexistantes), valeurs nulles, formats invalides,

date trop ancienne ou dans le futur, nombre trop grand, nombre de caractères plus grand que celui prévu en DB, ...

Il y a des tests des combinaisons d'erreurs.

Il y a des tests de données erronées dans un input.

Il y a des tests de recherche : aucune donnée ne correspond aux critères, trop de données correspondent aux critères (ne pas afficher plus de x données dans un tableau).

ETC.

Exemple de cas d'erreur isolée : création d'un utilisateur qui n'entre pas son prénom → le message d'erreur apparaît « vous devez introduire votre prénom ».

Formalisme :

Ne pas mettre de copie d'écran, ça serait vite non maintenable (cf. document de tests). Voir exemple TESTFONCT_0001 CreationEntreprise.pdf dans les slides.