

Syllabus

REQB®

Professionnel Certifié en Ingénierie des Exigences

Niveau fondation



**Requirements
Engineering**
Qualifications Board

Version 2.2

2015

Toute remarque ou demande de correction concernant la version française de ce document peut être transmise à traductions@cftl.fr

Le copyright ® de cette édition du syllabus est détenu dans toutes les langues par le
Global Association for Software Quality. GASQ

Historique des changements

Version	Date	Comment
0.1	Apr. 17, 2006	Première version du syllabus; création d'une structure de base pour le syllabus
0.2	Jul. 20, 2006	Extensions à la version 0.1
0.3	Sep. 4, 2006	Autres extensions et révisions de la version 0.1
0.4	Oct. 10, 2006	Version révisée 0.3
0.5	Dec. 15, 2006	Version révisée 0.4
0.6	Feb. 7, 2007	Version révisée complètement 0.5
0.7	Apr. 10, 2007	Version révisée pour revue
0.8	Jun. 15, 2007	Version Alpha
0.9	Sep. 1, 2007	Version Beta
1.0	Jan. 15, 2008	Version diffusée 1.0
1.1	May 29, 2008	Mise à jour version 1.1
1.2	Jul. 1, 2008	Mise à jour version 1.2
1.3	May 15, 2011	Mise à jour version 1.3
1.4	Jun. 30, 2013	Modification suite à la revue de la version 1.4
1.5	Aug. 19, 2013	Modifications : modifications et réduction des LOs, ajout de mappage entre les BOs et les LOs, changement de la structure du chapitre 3 - RE
1.6	Oct. 20, 2013	Réduction du nombre de BOs, modification de la classification des exigences, modification de la définition de modèle conceptuel, ajout de plus d'exigences concernant les exercices, réduction de la liste de la bibliographie, incorporation des commentaires de la revue de la version 1.5
1.7	Nov. 01, 2013	Modifications mineures suites au dernier cycle de révision

2.0	Dec. 01, 2013	Version finale
2.1	Feb 09, 2014	Modifications mineures
2.2	Nov 17, 2015	Modifications mineures

Idée principale

Le thème central pour l'élaboration de ce syllabus a été la croissance constante de la complexité des systèmes et des logiciels et notre dépendance à ces derniers. Ceci implique une exigence importante d'absence d'erreurs dans le logiciel, le matériel et les autres produits. Le « Requirements Engineering Qualifications Board » (REQB) a donc décidé de créer des standards internationaux uniformes dans le domaine de l'Ingénierie des Exigences. Les standards sont comme les langages, ce n'est que si vous les comprenez que vous serez en mesure de travailler efficacement. Pour créer un langage uniforme dans le domaine important qu'est l'Ingénierie des Exigences, des experts internationaux se sont réunis au sein du REQB et ont développé ce syllabus.

Remerciements

Ce document a été produit par l'équipe du Requirements Engineering Qualifications Board Working Party Foundation Level (Edition 2013) : Karolina Zmitrowicz (chair), Sergey Kalinov, Beata Karpinska, Andrey Konushin, Salvatore Reale, Alexander Lindner, Ingvar Nordström, Alain Ribault, Eric Riou du Cosquer.

L'équipe remercie tous les comités nationaux pour leurs suggestions et apports.

Table des matières

Introduction.....	7
1 Introduction aux Exigences.....	14
1.1 Le Concept d’Exigence	16
1.1.1 Le concept de problème, de solution et de produit.....	16
1.1.2 Définition et classification des exigences	16
1.1.3 Attributs des exigences.....	18
1.1.4 Qualité des exigences	20
1.1.5 Ingénierie des Exigences, gestion et développement des exigences	23
1.2 Standards et Normes	25
2 Contexte de l’Ingénierie des Exigences	27
2.1 Contexte de l’Ingénierie des Exigences.....	28
2.2 Processus Connexes	30
3 Processus d’Ingénierie des Exigences.....	32
3.1 Introduction au processus d’Ingénierie des Exigences	33
3.2 Processus Générique d’Ingénierie des Exigences	34
3.3 Rôles et Responsabilités.....	39
4 Gestion des Exigences.....	42
4.1 Introduction à la Gestion des Exigences	44
4.2 Gestion de Projet et Gestion du Risque.....	45
4.2.1 Introduction	45
4.2.2 Problèmes habituels dans la Gestion des Exigences	46
4.2.3 Gestion des Risques.....	46
4.3 Traçabilité des Exigences	49

4.3.1	Instabilité des Exigences	49
4.3.2	Cycle de Vie des Exigences	49
4.3.3	Traçabilité des Exigences	50
4.4	Gestion de Configuration et Gestion du Changement.....	52
4.4.1	Introduction	52
4.4.2	Gestion de Configuration.....	52
4.4.3	Gestion du Changement	53
4.4.4	Comité de Contrôle des Changements	54
4.5	Assurance Qualité	57
4.5.1	Introduction	57
4.5.2	Facteurs influençant le niveau de qualité attendu	57
4.5.3	Mesure de la qualité des exigences.....	58
5	Développement des Exigences	61
5.1	Introduction au Développement des Exigences	63
5.2	Elicitation des Exigences	64
5.3	Analyse des Exigences.....	78
5.3.1	Introduction	78
5.3.2	Estimation des coûts (charge)	79
5.3.3	Priorisation.....	82
5.3.4	Accord sur les exigences	83
5.3.5	Modélisation de la solution	84
5.4	Spécification des Exigences.....	89
5.5	Vérification et Validation des Exigences	94
6	L'Ingénierie des Exigences dans les Modèles	96
6.1	Approches de Développement et de Maintenance des modèles.....	97
6.2	Modèles de maturité	103
7	Support d'Outils	105
7.1	Avantage des Outils	106

7.2	Catégories d'Outils	107
7.3	Sélection des Outils	108
8	Tables / Illustrations	110
9	Standards	111
10	Livres et publications	112
11	Index	114

Introduction

Objectif de ce syllabus

Ce syllabus définit le niveau fondation du programme de formation pour devenir un professionnel certifié REQB de l'Ingénierie des Exigences (REQB Certified Professional for Requirements Engineering - CPRE). REQB a développé ce syllabus en collaboration avec le GASQ (Global Association for Software Quality). Le périmètre du programme REQB couvre le processus d'Ingénierie des Exigences pour tout type de produits IT pouvant inclure du logiciel, du matériel, des services ainsi que les exigences métier et la documentation associée.

REQB fournit ce syllabus :

1. Aux comités nationaux, pour le traduire dans leur langue locale et accréditer des fournisseurs de formations. La traduction peut inclure l'adaptation du syllabus aux besoins particuliers de langue et la modification des références (livres et publications) pour s'adapter aux publications locales.
2. Aux comités d'examen, pour leur permettre de créer des questions d'examen dans leur langue locale adaptée aux objectifs d'apprentissage (LOs) définis dans ce syllabus.
3. Aux fournisseurs de formations, cherchant l'accréditation leur permettant d'être reconnus comme fournisseur de formation REQB, pour produire des supports pédagogiques. Tous les domaines de ce syllabus doivent être incorporés dans les supports de formation de manière correspondante.
4. Aux candidats à la certification, comme matériel de préparation à l'examen de certification (en tant que partie d'un support de formation accrédité ou indépendamment).
5. Aux communautés internationales de l'Ingénierie des Exigences, pour promouvoir la profession d'ingénieur des exigences.

Examen

L'examen pour devenir un professionnel certifié en Ingénierie des Exigences niveau fondation est basé sur ce syllabus. Ainsi toutes les sections de ce programme peuvent être abordées au travers de questions de l'examen. Les questions d'examen ne sont pas nécessairement tirées d'une section individuelle ; une question peut se référer à plusieurs sections.

L'examen est composé de questions à choix multiple.

Les passages d'examen peuvent être tentés après avoir assisté à des formations accréditées ou en tant que candidat libre (sans un cours préalable). Vous trouverez des informations détaillées quant

aux temps d'examen sur le site web du REQB (www.reqb.org), sur le site web du GASQ (www.gasq.org) ou sur le site web de votre fournisseur d'examen local dans votre pays.

Accréditation

Les fournisseurs de cours REQB niveau fondation pour la certification du professionnel de l'Ingénierie des Exigences doit être accrédités par le Global Association for Software Quality. Leurs experts passent en revue la documentation du fournisseur de formation afin de contrôler l'exactitude et la conformité avec le contenu et les objectifs d'apprentissage (LOs : learning Objectives) du syllabus. Un cours est accrédité lorsqu'il est considéré être conforme au syllabus. À l'issue d'un tel cours, un examen de Certified Professional for Requirements Engineering (examen CPRE) peut être effectué par un institut de certification indépendant (selon la norme ISO 17024).

Les fournisseurs de formation accrédités peuvent être identifiés grâce au logo officiel des fournisseurs de formation accrédités REQB.

Internationalité

Ce syllabus a été développé en collaboration entre plusieurs experts internationaux. Le contenu de ce syllabus peut donc être considéré comme un standard international. Le syllabus permet ainsi de former et de faire passer des examens internationalement au même niveau.

Bénéfices métier

Les objectifs, avantages et principaux points du programme REQB niveau fondation de professionnel certifié en Ingénierie des Exigences sont présentés dans la table ci-dessous :

Apports de l'Ingénierie des Exigences	
Obtenir de nouvelles qualifications clés	N'importe quelle solution logicielle, matérielle ou de service est basée sur les exigences des parties prenantes et a pour but de satisfaire des besoins métier spécifiques. Pour pouvoir livrer une solution conforme aux besoins du groupe cible, une Ingénierie des Exigences appropriée est nécessaire. Gérer les exigences et les développer selon la conception de la solution est très important, mais pour atteindre pleinement le succès et livrer la meilleure solution, il est également nécessaire de prendre en considération la qualité. Tous ces aspects sont couverts dans le niveau fondation.
Augmenter la satisfaction de vos clients	La satisfaction client est atteinte lorsqu'ils reconnaissent la solution comme répondant à leurs attentes et besoins. Améliorer l'Ingénierie des Exigences minimise les divergences entre les attentes et la perception de la conformité. Une qualité élevée du produit final renforce également la fidélité du client.
Minimisez les coûts de développement et de suivi	Une Ingénierie des Exigences appropriée minimise les risques projet et produit et évite les coûts liés aux reprises résultant, par exemple, de contradictions entre les attentes du client et l'efficacité de la solution développée. Des exigences bien gérées réduisent également les coûts des actions d'améliorations ou correctives.
Avantage compétitif	L'Ingénierie des Exigences aide à livrer de meilleurs produits ou services, répondant à tous les besoins et attentes métier. De meilleurs produits ou services aident à obtenir la confiance et à fidéliser les groupes cibles désirés et augmenter l'avantage compétitif.

Objectifs du programme de niveau Fondation	
Les exigences	Comprendre les concepts de base relatifs aux exigences, leur classification et niveaux d'abstraction. Expliquer la signification d'attribues des exigences et le rôle des critères qualité.
Processus d'Ingénierie des Exigences	Expliquer le processus d'Ingénierie des Exigences, ses activités, acteurs et livrables, les principes les plus importants et les bonnes pratiques pour développer et gérer les exigences de produits logiciel/matériel/service.
Standards, normes et réglementations juridiques	Vue d'ensemble des standards, normes et réglementations juridiques les plus importantes et leur effet sur l'Ingénierie des Exigences.
L'Ingénierie des Exigences dans le développement et la maintenance de modèles	Expliquer les principes d'ajustement du processus générique de l'Ingénierie des Exigences au processus spécifique des modèles.
Outils et techniques	Expliquer l'utilisation et les avantages des techniques permettant l'identification des exigences ; la modélisation de problème et de solution ; l'assurance qualité et techniques de contrôle ; les outils de support du processus d'Ingénierie des Exigences.
Exercices	Écriture des exigences correctes ; contrôle qualité des exigences ; identification et analyse des exigences.

Tableau 1: Apports de l'Ingénierie des Exigences et objectifs du programme de niveau fondation, ses avantages et principaux points

Le programme de niveau fondation de certification du professionnel de l'Ingénierie des Exigences du REQB est approprié pour toutes les personnes impliquées dans le développement et la maintenance de solution de produit/métier, à savoir : les analystes métier et système, les équipes marketing, les concepteurs de matériel/logiciel, les concepteurs d'IHM, les chefs de projet, les représentants du client, le personnel technique et de maintenance, les auditeurs IT, les représentants de l'assurance qualité.

Le but principal du programme de niveau fondation est de fournir une terminologie et une compréhension commune des concepts clés liés au processus d'Ingénierie des Exigences. Une base de connaissances, fournie dans le programme de niveau fondation, soutient des définitions communes et des concepts liés à l'Ingénierie des Exigences et explique le processus d'Ingénierie des

Exigences au travers de ses livrables. Le programme est basé sur des standards et les bonnes pratiques généralement admises.

Objectifs de connaissance

Les objectifs de connaissance de ce programme ont été divisés en différents niveaux cognitifs de connaissance (les niveaux K). Ceci permet au candidat de reconnaître "le niveau de connaissance" de chaque point.

Chaque section de ce programme est associée un niveau cognitif :

- K1 – Connaître/avoir la compétence : connaissance de détails précis tels que des termes, définitions, faits, données, règles, principes, théories, caractéristiques, critères, procédures. Les étudiants peuvent se rappeler et exprimer les connaissances.
- K2 – Comprendre : les étudiants peuvent expliquer ou résumer des faits avec leurs propres mots, fournir des exemples, comprendre des contextes, interpréter des tâches.
- K3 – Savoir appliquer : les étudiants peuvent appliquer leurs connaissances dans de nouvelles situations spécifiques, par exemple, en appliquant certaines règles, méthodes ou procédures.

Résultats métier (BO - Business Outcomes)

A l'issue du niveau fondation de certification du professionnel de l'Ingénierie des Exigences du REQB, une personne est en mesure de :

- BO01 Communiquer les concepts fondamentaux d'Ingénierie des Exigences, les exigences et leur rôle dans le cycle de vie d'un produit.
- BO02 Faire prendre conscience de la signification du processus d'Ingénierie des Exigences et de ses livrables.
- BO03 Communiquer les activités principales et le but du processus d'Ingénierie des Exigences, de la gestion des exigences et du développement des exigences.
- BO04 Communiquer les aptitudes de base et compétences exigées d'un ingénieur des exigences.
- BO05 Communiquer la signification d'Ingénierie des Exigences pour le succès du développement et/ou de la maintenance d'un projet.

- BO06 Communiquer l'utilisation possible de diverses techniques pour l'élicitation des exigences.
- BO07 Communiquer l'importance de la traçabilité et de la priorisation ainsi que leurs significations pour le processus de l'Ingénierie des Exigences.
- BO08 Communiquer l'utilisation de la modélisation pour créer une solution métier pour un problème métier donné.
- BO09 Communiquer l'importance de l'assurance qualité et le contrôle de qualité pour le processus d'Ingénierie des Exigences.
- BO10 Communiquer les similitudes et les différences du processus d'Ingénierie des Exigences entre les modèles communs en rapport avec le processus de développement/maintenance.

Niveau de détail

Le syllabus niveau fondation de certification du professionnel de l'Ingénierie des Exigences du REQB est destiné à soutenir la cohérence internationale de la formation et de l'examen. Ce programme comprend les composants suivants pour atteindre ce but :

- des objectifs généraux d'instruction décrivant l'intention du programme de niveau fondation de la certification REQB,
- des objectifs d'apprentissage pour chaque domaine de connaissances décrivant l'objectif de résultat de connaissances cognitif du cours et les qualifications que le participant doit atteindre,
- une liste d'informations à enseigner, incluant une description et les références à des sources additionnelles telles que la littérature technique, les normes ou les standards, si nécessaire,
- une liste des termes que les participants doivent pouvoir se rappeler et comprendre. Les termes individuels sont décrits en détail dans le « glossaire des termes standard utilisés en Ingénierie des Exigences » de REQB.

Le contenu du syllabus n'est pas une description exhaustive du domaine de connaissances de l'Ingénierie des Exigences. Il couvre le périmètre et le niveau de détail approprié du niveau fondation de la certification.

Organisation du Syllabus

Le Syllabus est composé de sept chapitre majeurs.

L'entête principal de chaque chapitre indique le sujet couvert par celui-ci et spécifie le temps minimal qu'un cours accrédité doit passer sur le chapitre.

Les objectifs d'apprentissage à satisfaire pour chaque chapitre sont inscrits au début du chapitre.

Dans chaque chapitre il y a un certain nombre de sections. Chaque section nécessite un temps minimal qu'un cours accrédité doit passer sur cette section. Les subdivisions qui n'ont pas de temps associé sont incluses dans le temps de la section.

Par exemple :

1 Introduction aux exigences	90 minutes
-------------------------------------	-------------------

Objectifs d'apprentissage du niveau fondation d'Ingénierie des Exigences.

Les objectifs identifient ce que vous pourrez démontrer à l'issue de chaque module.

L'en-tête principal indique qu'un minimum de 90 minutes doit être prévu pour enseigner le thème du chapitre.

Dans chaque chapitre il peut y avoir plusieurs sections chacune avec des objectifs d'apprentissage spécifiques. Le contenu d'une section spécifique doit permettre de réaliser les objectifs d'apprentissage appropriés.

1 Introduction aux Exigences	110 minutes
-------------------------------------	--------------------

Termes

Contrainte métier, problème métier, engagement, contrainte, criticité, exigence fonctionnelle, exigence non-fonctionnelle, attributs qualité des exigences, priorité, produit, exigences produit, exigences, exigences de développement, Ingénierie des Exigences, gestion des exigences, solution, système, contrainte technique, validation, vérification

Objectifs d'apprentissage pour le niveau fondation de l'Ingénierie des Exigences

Les objectifs identifient ce que vous pourrez démontrer à l'issue de chaque module.

1.1 Le Concept d'Exigence

1.1.1 Le concept de Problème, de Solution et de Produit

LO-1.1.1 Expliquer les concepts de problème, solution et produit (K2)

1.1.2 Définition et Classification des Exigences

LO-1.1.2 Expliquer le concept d'exigence et leur classification (c.à.d. exigences produit et contraintes) (K2).

LO-1.1.3 Expliquer les différences entre les différents niveaux d'exigences (par exemple, les exigences métier, les exigences de solution (système) et les exigences produit (composant)) ainsi que leur signification pour l'Ingénierie des Exigences (K2).

1.1.3 Attributs Communs des Exigences

LO-1.1.4 Expliquer la notion d'attributs de description d'une exigence (par exemple, l'engagement, la priorité et la criticité) (K2).

1.1.4 Qualité des Exigences

- LO-1.1.5 Comprendre les principaux problèmes concernant les exigences et expliquer le rôle de la validation, de la vérification et des critères qualité dans l'amélioration de la qualité des exigences (K2).

1.1.5 Ingénierie, Gestion et Développement des Exigences

- LO-1.1.6 Comprendre le besoin d'une vision processus pour l'Ingénierie des Exigences ainsi que son impact, afin d'éviter des erreurs dans l'Ingénierie des Exigences (K2).

1.2 Standards et Normes

- LO-1.2.1 Expliquer les possibles bénéfices et l'application de standards pour l'Ingénierie des Exigences (K2).

1.1 Le Concept d'Exigence

90 minutes

Pour comprendre la discipline d'Ingénierie des Exigences et ses principaux entrants et sortants, il est important de cerner le contexte et de présenter les termes et définitions les plus importants.

L'intégralité des termes et définitions peut être trouvée dans le « Glossaire des termes de l'Ingénierie des Exigences ».

Le but de ce chapitre est de fournir les concepts de base d'Ingénierie des Exigences et comment ils sont liés à une exigence.

1.1.1 Le concept de problème, de solution et de produit

Les principaux entrants du processus d'Ingénierie des Exigences sont des problèmes métier, des besoins métier et/ou des objectifs métier. Un problème métier est la description de ce qu'un client souhaite faire pour réaliser ou améliorer ses processus métier. Le problème décrit ou aide à décrire les besoins métier d'un client.

Un des objectifs principaux de l'Ingénierie des Exigences doit être de définir des solutions métier pour des problèmes donnés métier, des besoins métier ou des objectifs. Une solution est la réponse aux besoins d'un client. Ces besoins sont habituellement exprimés sous forme d'exigences clients et sont des entrants à l'Ingénierie des Exigences. Une solution peut être un système ou un composant de système, une fonctionnalité nouvelle ou changée, une configuration changée, une amélioration de processus, etc. En général, une solution est un raffinement d'une exigence de plus haut niveau.

Le résultat d'un processus de réalisation est appelé produit.

[Pour les organismes de formation : expliquer le concept de problème, de solution et de produit avec un exemple réel de projet de développement ou de maintenance.]

1.1.2 Définition et classification des exigences

Selon IEEE 610, une exigence est définie comme :

- 1) Une condition ou une capacité requise par un utilisateur pour résoudre un problème ou atteindre un objectif.
- 2) Une condition ou capacité qui doit être tenue ou possédée par un système ou composant pour satisfaire un contrat, standard, spécification ou autre document imposé formellement.
- 3) Une représentation documentée d'une condition ou capacité telle que 1) ou 2).

Un des principaux objectifs des exigences est d'exprimer les besoins utilisateurs et des attentes quant à la solution prévue. Les exigences servent aussi de base à l'évaluation, la planification, l'exécution et le suivi d'activités projet et sont souvent une des composantes des engagements de service, commandes ou contrats. Les exigences ne définissent pas seulement ce qui doit être fait mais elles établissent les frontières de la solution, le périmètre de livraison et des services contractuels.

En général, les exigences peuvent être classifiées en exigences produit et contraintes.

Les exigences produit décrivent des caractéristiques, des fonctionnalités, des qualités et services requises par un produit donné et sont composées d'exigences fonctionnelles et non fonctionnelles. Les exigences fonctionnelles décrivent la fonction (traitement) du produit, alors que les exigences non fonctionnelles décrivent ce que l'on appelle les attributs qualité des produits (aussi mentionnés comme objectifs qualité).

Les différences entre exigences fonctionnelles et non fonctionnelles peuvent être exprimées par les principes suivants :

- Les exigences fonctionnelles décrivent ce *QUE* la solution fait.
- Les exigences non fonctionnelles décrivent *COMMENT* la solution exécute ses fonctionnalités

Les contraintes sont des limitations du processus d'ingénierie, du fonctionnement du système ou de son cycle de vie. Il y a deux types de contraintes : les contraintes métier et les contraintes techniques.

- Les contraintes métier expriment les limitations de flexibilité du projet à mettre en œuvre la solution attendue (par exemple des restrictions financières ou temporelles, des limites du nombre de ressources disponibles, des compétences de l'équipe projet, d'autres restrictions organisationnelles, des lois de domaine, des normes spécifiques et des règlements).
- Les contraintes techniques représentent toutes les restrictions liées à l'architecture de la solution (par exemple les plateformes matérielles et logicielles, les langages et technologies de programmation, le logiciel qui doit être utilisé, la taille de la base de données, l'utilisation de ressources, la taille et timing des messages, le dimensionnement logiciel, le nombre maximal et la taille des fichiers, les enregistrements et éléments de données).

Dans le cas des organisations utilisant un produit, une exigence peut être externe ou interne – certaines exigences peuvent venir des clients et d'autres de l'organisation elle-même. Les exigences provenant de différentes sources devraient être analysées ensemble.

Les exigences peuvent être classées en types selon qu'elles traitent les différents aspects du produit ou de son processus de développement. Elles peuvent être classées dans des niveaux d'abstraction représentant les différents niveaux de détail d'une exigence allant des besoins métier de haut niveau aux exigences détaillées liées aux fonctionnalités logicielles ou système voire aux éléments de conception de la solution. (Ex. prototypes d'écran, clavier de téléphone mobile).

Certains des niveaux d'abstraction courants des exigences incluent les éléments suivants :

- Les exigences métier
 - Exigences de haut niveau définissant ce que le métier veut réaliser mais pas comment le mettre en œuvre. Ces exigences expriment les désirs des utilisateurs, les besoins et attentes. Elles sont souvent référencées comme des exigences métier de haut niveau ou client.
- Les exigences solution/système
 - Raffinement des exigences métier décrivant la solution qui est une des façons possibles de répondre aux exigences métier. La solution peut contenir des exigences non liées à l'IT pour les modifications de processus ou les changements organisationnels / rôles. Les exigences solution / système sont l'expression des exigences métier en termes plus techniques qui peuvent être utilisés pour des décisions de conception.
- Les exigences produit/composant
 - Fonctions et caractéristiques de la solution. Une spécification complète d'un composant de produit y compris l'adaptation, la forme, la fonction, la performance etc. Ce niveau d'abstraction est très souvent absent car il est considéré comme une partie de la conception de la solution.

Lorsque vous travaillez avec des exigences sur les différents niveaux, il est important de définir et de maintenir la traçabilité (voir la section 4.3 Traçabilité des exigences). La traçabilité est un lien entre les objets projet sur les différents niveaux, par exemple, entre les exigences métier et les exigences solution ou entre les exigences et les cas de tests associés.

[Pour les organismes de formation : fournir des exemples de besoins, de différents types et niveaux d'abstraction exigences.]

1.1.3 Attributs des exigences

Les attributs les plus importants pour une exigence de haut niveau sont les suivants :

Niveau fondation

- Engagement
- Priorité
- Criticité

L'engagement est le degré d'obligation pour satisfaire à l'exigence. Il est généralement défini par des mots clés attribués aux exigences de haut niveau (« doit », « devrait », « pourrait », « peut ») [d'après la norme ISO / IEC / IEEE 29148:2011, anciennement IEEE 830]. Dans certains cas, le mot clé "ne doit pas" est utilisé pour exprimer des aspects que la solution ne doit pas faire. Les mots clés "devraient", "pourraient", "peuvent" se rapportent habituellement aux exigences métier avant l'accord. Après acceptation et référencement, les mots clés des exigences doivent déterminer le degré d'obligation en termes plus stricts : « fera » ou « doit ».

Le niveau d'obligation d'une exigence peut être exprimé aussi en utilisant la notation de MOSCOW (doit avoir, devrait avoir, aurait pu, n'aura pas cette fois mais il aura à l'avenir). Une fois que le fournisseur de la solution et le client parviennent à un accord, l'engagement des exigences est obtenu par les participants du projet.

Les exigences peuvent évoluer tout au long du projet. Comme les exigences évoluent, obtenir l'engagement des participants sur les exigences modifiées garantit l'engagement sur les changements qui en résultent dans les plans projet, les activités et les réalisations.

Une des conséquences les plus importantes de l'engagement est la responsabilité pour les produits finaux. Il peut y avoir des responsabilités juridiques liées à la qualité du produit. Les responsabilités juridiques sont souvent liées à une exigence spécifique (par exemple, une exigence environnementale d'une centrale nucléaire ou une exigence de sûreté de fonctionnement pour un avion) qui doit être satisfaite par le produit livré. Les responsabilités peuvent également concerner des défauts dans le produit. L'engagement est une façon d'exprimer le respect des exigences juridiques.

Les responsabilités juridiques devraient être définies dans le contrat entre le fournisseur et le client. Certaines d'entre elles peuvent également être nécessaires pour répondre aux exigences contractuelles ou légales ou des normes spécifiques à l'industrie.

L'attribut suivant d'une exigence – la priorité - exprime l'importance / urgence d'une exigence. Selon [SWEBOOK], en général, plus la priorité est haute, plus l'exigence est essentielle pour atteindre les objectifs globaux du produit. Habituellement classée sur une échelle de points fixes tels qu'obligatoire, hautement souhaitable, désirable ou optionnelle, la priorité doit souvent être mise en balance avec le coût de développement et de mise en œuvre.

La criticité d'une exigence est le résultat d'une évaluation du risque des dommages qui se produiraient si cette exigence n'est pas satisfaite. La criticité est exprimée en niveaux ; plus le niveau est élevé, plus les conséquences sont graves en cas de défaut de fonctionnement.

Dans certains cas, les attributs d'exigences peuvent être également exprimés sous la forme du modèle de Kano. Ce modèle met l'accent sur la différenciation des caractéristiques du produit de la manière suivante :

- Les seuils ou attributs de base - ce sont les caractéristiques que le produit doit avoir afin de répondre aux besoins des clients.
- Les attributs de performance - une compétence, des connaissances, une capacité ou caractéristique comportementale qui est associée à la manière dont le produit offrira ses aptitudes à l'utilisateur. Les attributs de performance améliorent la satisfaction du client.
- Les attributs d'excitation – les attributs non prévus par le client (parfois appelés « besoins inconnus ») qui sont à découvrir par le fournisseur. Ils éclairent le client et peuvent fournir un avantage concurrentiel significatif.

Le modèle de Kano peut être utilisé pour exprimer à la fois la priorité et l'engagement.

En plus des caractéristiques décrites ci-dessus, les exigences peuvent être classées de la manière suivante [ISO / IEC / IEEE 29148:2011, anciennement IEEE 1233]:

- Identification
- Faisabilité
- Risque
- Origine
- Type

[Pour les organismes de formation : expliquer les attributs courants d'une exigence et produire des exemples démontrant leur signification.]

1.1.4 Qualité des exigences

L'Ingénierie des Exigences est l'un des facteurs de succès les plus importants de tout projet. Les exigences sont la base de la poursuite des travaux ; par conséquent, il est très important d'assurer la meilleure qualité possible des exigences et des autres produits du processus d'Ingénierie des Exigences. Les plus importants problèmes à prendre en considération lorsque l'on travaille avec les exigences sont les suivants :

- Le manque de clarté des objectifs métier pour les exigences ou le projet lui-même
- Les problèmes de communication (résultant souvent de la langue et / ou des barrières de connaissances, y compris le manque de connaissances du domaine métier), les formulations vagues (résultant souvent d'une définition insuffisante ou non mesurable d'une exigence ou l'absence de glossaire commun)

- La volatilité des exigences (souvent causée par des objectifs métier peu clairs pour les exigences ou le projet lui-même)
- La mauvaise qualité des exigences (voir les critères de qualité pour les exigences, ci-dessous)
- Le plaqué or (ajout de fonctionnalités supplémentaires qui ne sont pas vraiment nécessaires)
- L'implication insuffisante des parties prenantes
- Les parties prenantes manquantes (souvent résultant d'un processus d'Ingénierie des Exigences défaillant)
- La planification inexacte (souvent causée par des objectifs métier peu clairs ou manquants pour les exigences ou le projet lui-même)
- Les exigences insuffisantes ou des spécifications de solution (avec pour résultats des écarts, des manques ou des exigences fragmentées)

Certains des problèmes mentionnés ci-dessus peuvent être résolus tout simplement en appliquant des critères de qualité aux exigences. Selon [Wiegers], les critères de qualité communs pour les exigences définissent que chaque exigence doit être :

- Exacte - l'exigence doit décrire avec précision les caractéristiques à fournir. Le point de référence pour évaluer l'exactitude est la source de l'exigence (par exemple, les clients ou une exigence du système de niveau supérieur).
- Réalisable - l'exigence doit être implémentable dans les aptitudes et/ou les limites connues du système et de l'environnement.
- Nécessaire - l'exigence devrait documenter ce que le client (ou d'autres parties prenantes) a vraiment besoin et ce qui est nécessaire pour accomplir une obligation externe, une interface ou une norme précise.
- Priorisée - l'exigence doit avoir une priorité assignée indiquant combien elle est essentielle pour une version d'un produit particulier.
- Non ambiguë - l'exigence ne doit être interprétée que dans un sens. Différents lecteurs d'une exigence doivent avoir la même interprétation et compréhension d'une exigence.
- Vérifiable - l'exigence doit pouvoir être vérifiée si elle est correctement mise en œuvre.
- Unicité - l'exigence ne contient pas plusieurs exigences ; ce qui implique une granularité suffisante pour spécifier une seule exigence.
- Indépendante de la conception (mise en œuvre) - l'exigence doit décrire « ce qui doit être fait » et pas « comment le faire ». Le contenu d'une exigence ne devrait pas prescrire ou impliquer ses détails de mise en œuvre.

Les critères de qualité peuvent être appliqués non seulement à une seule exigence, mais peuvent être aussi utilisés pour la spécification des exigences.

Les principaux critères qualité pour les spécifications des exigences indiquent que la documentation doit être :

- Complète – aucune exigence ou information nécessaire ne devrait être absente de la spécification des exigences. L'exhaustivité est également exprimée comme une caractéristique souhaitée d'une exigence individuelle et niveau de détail.
- Cohérente – les exigences décrites dans la spécification ne peuvent pas entrer en conflit avec d'autres exigences produit ou de niveau supérieur (système ou métier).
- Modifiable - la spécification doit permettre l'introduction de changements dans les exigences. Un historique des modifications apportées à chaque exigence doit être maintenu.
- Traçable - chaque exigence devrait pouvoir être rattachée à la source (par exemple, une exigence de haut niveau système, un cas d'utilisation, ou la déclaration d'un client) et les artefacts de mise en œuvre connexes (par exemple, des éléments de conception, le code source et des cas de test).

[Pour les organismes de formation : expliquer les critères qualité courants des exigences et fournir des exemples d'exigences répondant ou ne répondant pas aux critères qualité.]

Assurer le niveau requis de qualité des exigences ou d'autres produits du processus d'Ingénierie des Exigences peut être pris en charge par la validation et la vérification.

Selon le CMMI, les activités de validation démontrent qu'un produit ou un composant de produit remplit sa fonction lorsqu'il est placé dans son environnement prévu. Alors, vous savez que "vous avez construit le bon produit". Les clients fournissent souvent des descriptions ou des exigences produit de manière vague et la validation aide à comprendre ce qui est nécessaire (en utilisant des outils tels que des scénarios, cas d'utilisation, prototypage, etc.).

La validation est généralement menée avec le soutien d'un client ou de parties prenantes sur le site client et vise à confirmer que les exigences ou les spécifications des exigences décrivent précisément ce que les clients ont besoin.

Selon le CMMI, la vérification fournit des points de contrôle auxquels les livrables sélectionnés ou les produits intermédiaires sont vérifiés pour s'assurer qu'ils répondent à leurs exigences. Les activités de vérification se concentrent sur la confirmation incrémentale de la mise en œuvre des exigences ; ils permettent une confirmation tôt et de manière continue que l'on construit le produit attendu.

La vérification est une comparaison d'un produit intermédiaire à ses spécifications. Cela permet de déterminer si le produit a été développé correctement et si les spécifications qui ont été déterminées au cours de la phase précédente ont été satisfaites.

Les techniques les plus courantes pour la vérification sont les commentaires, les analyses statiques et les tests dynamiques. Pour la validation, les revues et les tests dynamiques sont utilisés.

La différence entre la validation et la vérification peut être exprimée de la manière suivante :

- Vérification - avons-nous créé correctement le produit ?
- Validation - avons-nous créé le bon produit ?

1.1.5 Ingénierie des Exigences, gestion et développement des exigences

L'Ingénierie des Exigences (RE) peut être définie comme une sous-discipline de l'ingénierie des systèmes, axée sur la détermination, l'élaboration et la gestion des exigences des systèmes matériels et logiciels [SWEBOK, Sommerville]. Selon CMMI, l'Ingénierie des Exigences englobe la gestion des exigences et le développement des exigences.

Le but de la gestion des exigences est de gérer la référence d'un ensemble d'exigences de la solution retenue et d'assurer l'alignement entre ces exigences avec les plans projet et les réalisations.

La Gestion des Exigences constitue à la fois un cadre de travail pour l'Ingénierie des Exigences et les processus de support du processus de développement des exigences. L'autre rôle de la Gestion des Exigences est d'établir et de fournir des interfaces aux autres processus de développement et managériaux en relation avec l'Ingénierie des Exigences telles que la gestion de projet, la gestion des risques, la conception, la gestion de la configuration, la gestion du changement et l'assurance qualité.

Le Développement des Exigences est un ensemble d'activités, de tâches, de techniques et d'outils pour identifier, analyser, documenter et valider les différents niveaux d'abstraction des exigences. Il comprend le processus de transformation des besoins en exigences ainsi que le développement (raffinement des exigences de haut niveau) d'une solution pour ces exigences.

Un bon processus d'Ingénierie des Exigences est l'un des principaux facteurs de réussite de tout projet. Des exigences bien recueillies, analysées, documentées et gérées diminuent les risques du projet car elles constituent une base claire et comprise à la conception de la solution. Cependant, les apports de bonnes exigences et d'un processus d'Ingénierie des Exigences ne sont pas encore bien compris ou tout simplement négligés.

Les raisons de négliger l'Ingénierie des Exigences peuvent être les suivantes :

- Pression des délais et des coûts
- Orientation exclusive vers des résultats rapides
- Prise en compte des seules exigences fonctionnelles
- Manque de compréhension de l'importance de l'Ingénierie des Exigences pour la réussite d'un projet

Les conséquences d'un processus d'Ingénierie des Exigences pauvre peuvent être les suivantes :

- Exigences de faible qualité
- Exigences qui changent souvent au cours du développement du produit
- Exigences qui ne remplissent pas les critères d'acceptation et/ou ne fournissent pas une valeur pour le client ou d'autres parties prenantes
- Exigences ou contraintes manquantes

1.2 Standards et Normes

20 minutes

Beaucoup de règles et normes de processus sont utiles pour l'Ingénierie des Exigences. Certaines d'entre elles offrent des modèles de processus liés au développement de la solution, d'autres fournissent des lignes directrices pour la rédaction de différentes spécifications des exigences ou de classification des exigences.

L'un des principaux objectifs des normes est d'avoir un alignement national et international des produits et des processus. Les Standards normalisent les méthodes de développement et les livrables, fournissent une terminologie commune et facilitent la communication vers les équipes métier et techniques. Les standards et les normes les plus importants utilisés dans le schéma REQB sont énumérés ci-dessous.

Standards :

- ISO 9000 : La famille des normes ISO 9000 définit les exigences pour un système de gestion de la qualité et définit les concepts et principes de base d'un système de gestion de la qualité (SMQ).
- ISO / CEI 25000 (anciennement ISO 9126) : Il définit un modèle de qualité avec six grandes catégories (fonctionnalité, fiabilité, facilité d'utilisation, efficacité, maintenabilité, portabilité) qui peuvent être une base pour l'Elicitation, la Spécification ou la validation et la vérification des exigences.
- IEEE 610.12-1990 : Glossaire standard IEEE de terminologie du génie logiciel
- IEEE 830-1998 : IEEE pratique recommandée pour la spécification des exigences logicielles (également connue sous le nom pratiques recommandées pour SRS).
- IEEE 1233-1998 : Guide IEEE pour le développement de spécifications d'exigences système (également connu sous le Guide d'élaboration de SyRS).
- IEEE 1362-1998 : Guide IEEE pour la rédaction des définitions/concepts d'opération des technologies/systèmes d'information
- Remarque : IEEE 830, IEEE 1233 et IEEE 1362 ont été remplacées par la norme ISO / IEC / IEEE 29148: 2011
- SWEBOK : Le Guide des connaissances d'Ingénierie Logicielle (connu sous le nom Rapport technique ISO 19759). SWEBOK décrit les connaissances généralement reconnues pour le génie logiciel. Ses 10 domaines de connaissances résument les concepts de base et comprennent une liste de référence indiquant les informations détaillées. L'un des domaines est dédié à l'Ingénierie des Exigences.
- SEBOK : Le Guide des connaissances d'Ingénierie Système.

Normes de processus pour le développement :

- ISO 12207 : standard pour le processus du cycle de vie du logiciel.
- ISO 15288 : standard pour le processus du cycle de vie du système.

Les deux peuvent être utilisés pour supporter l'organisation du processus de développement de la solution.

Normes de processus d'évaluation et amélioration des processus :

- ISO 15504 Technologies de l'information - Évaluation des procédés, aussi connu comme SPICE (Détermination d'amélioration et de capacité du processus logiciel).
- CMMI. Capability Maturity Model Integrated.

Les deux soutiennent l'amélioration de processus et peuvent être utilisés pour l'évaluation de la maturité et l'amélioration des processus. Ils définissent les domaines clés de l'Ingénierie des Exigences.

[Pour les organismes de formation : expliquer pourquoi ces normes sont pertinentes et importantes lors de la définition d'un processus d'Ingénierie des Exigences]

2 Contexte de l'Ingénierie des Exigences

40 minutes

Termes

Analyse métier

Objectifs d'apprentissage pour le niveau fondation de l'Ingénierie des Exigences

Les objectifs identifient ce que vous pourrez démontrer à l'issue de chaque module.

2.1 Contexte de l'Ingénierie des Exigences

LO-2.1.1 Expliquer le contexte de l'Ingénierie des Exigences et comment elle fait partie du développement de la solution et de la maintenance (K2)

2.2 Processus Connexes

LO-2.2.1 Expliquer les relations et interactions entre l'Ingénierie des Exigences et les autres activités du projet de développement (K2)

2.1 Contexte de l'Ingénierie des Exigences	20 minutes
---	-------------------

L'Ingénierie des Exigences n'est pas mise en œuvre de façon isolée. Elle est en lien avec d'autres disciplines (figure 1) et doit être prise en compte tout au long du processus de développement d'une solution.

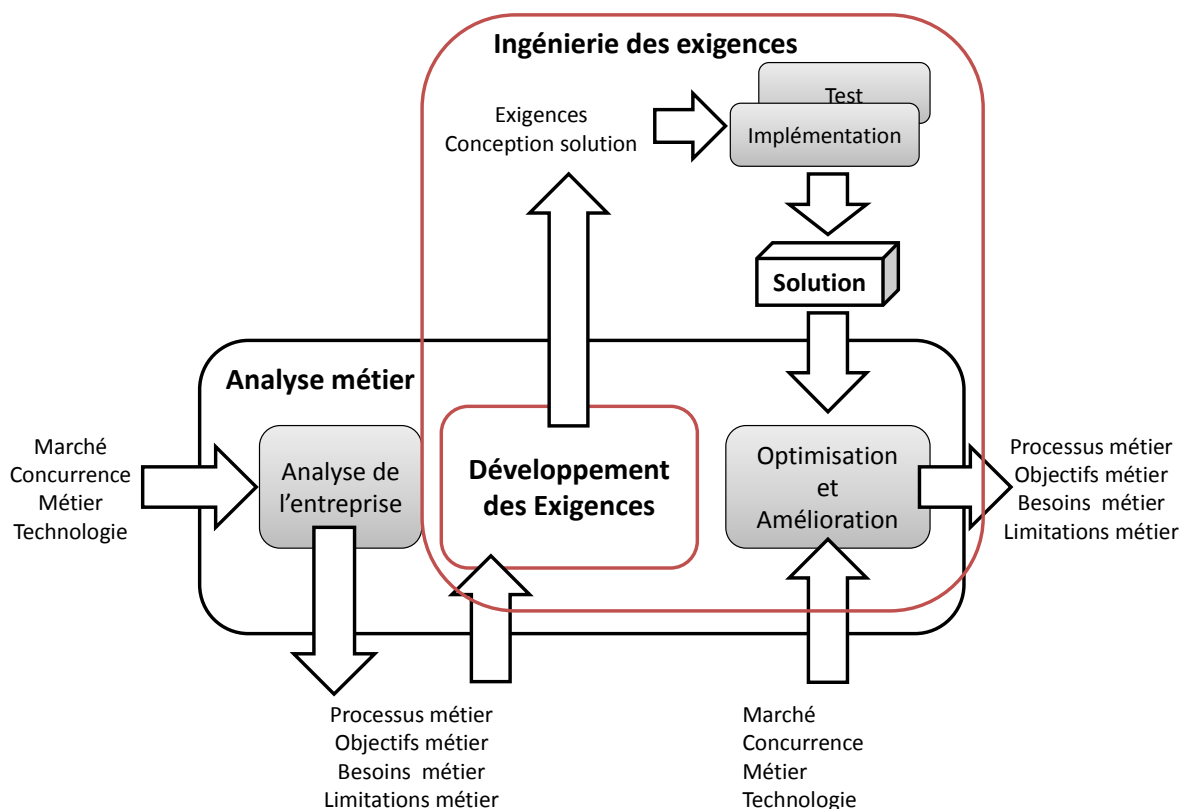


Figure 1 - Contexte de l'Ingénierie des Exigences

Le point d'entrée de l'Ingénierie des Exigences est l'analyse métier. Pour définir la meilleure solution associée à un problème métier, il est primordial de décrire correctement ce problème [IQ-BBA].

L'analyse métier est une discipline comprenant un ensemble d'activités, d'outils et de méthodes avec pour but de décrire les besoins, problèmes et objectifs métier, de déterminer les solutions appropriées pour satisfaire ces besoins et résoudre les problèmes métier posés. Ces solutions métier devraient inclure le développement de systèmes ou composants logiciels, des extensions de logiciels existants, l'amélioration de processus métier, des changements d'organisation, etc.

L'analyse métier décrit le problème métier à résoudre par une solution spécifique qui devient une entrée pour les prochaines activités d'Ingénierie des Exigences. Les besoins et objectifs métier définis pendant l'analyse métier sont développés en exigences pour la solution. Donc, l'Ingénierie des Exigences peut être vue comme une continuité ou une partie du processus d'analyse métier. La sortie de l'Ingénierie des Exigences est la description de la solution qui sera par la suite développée, testée et finalement fournie au client.

La solution implémentée reste rarement sans changement pendant une longue période de temps. Les cibles marché et commerciales changent, la technologie évolue constamment, de nouvelles connexions avec les processus métier apparaissent.

Presque tous les systèmes et toutes les solutions subiront des changements et verront la sortie de nouvelles versions.

Il est donc important que l'analyse métier et l'Ingénierie des Exigences prennent en compte aussi les activités de maintenance. C'est pourquoi l'analyse métier – et l'Ingénierie des Exigences – continuent pendant tout le cycle de vie de la solution. Après la livraison de la solution, l'analyse métier identifie des améliorations qui vont permettre d'augmenter les parts de marché, améliorer l'efficacité opérationnelle et gagner des parts concurrentielles. De nouveaux besoins métier apparaissent. Ces besoins métier sont développés par l'Ingénierie des Exigences pour proposer de nouvelles solutions, ou étendre des solutions existantes avec de nouvelles fonctionnalités.

[Pour les organismes de formation : expliquer le contexte de l'Ingénierie des Exigences avec des exemples et retours d'expérience. Fournir des exemples de relations entre l'analyse métier et l'Ingénierie des Exigences.]

2.2 Processus Connexes

20 minutes

L'Ingénierie des Exigences, et en particulier la partie appelée « Gestion des exigences », s'inscrit dans un plus large contexte et a de fortes relations avec les processus suivants :

Analyse métier : Comme expliqué dans la section précédente, l'analyse métier et l'Ingénierie des Exigences sont fortement liées.

- Gestion de projet - Le chef de projet devrait inclure des activités spécifiques aux exigences dans son plan de gestion de projet, allouer le temps et les ressources nécessaires et s'assurer qu'elles sont correctement effectuées. Les ingénieurs d'exigences peuvent avoir à produire un plan des exigences qui détaille les tâches requises pour compléter la spécification des exigences.
- Gestion des risques - Certaines exigences peuvent introduire des risques projets ou produits qui doivent être gérés via le processus de gestion des risques.
- Analyse et Conception - Les exigences sont une entrée obligatoire pour les activités d'analyse et de conception. La traçabilité entre les exigences et les éléments d'analyse et de conception est nécessaire.
- Gestion de Configuration et du changement - Les exigences devraient être gérées en configuration et suivies via le processus de gestion de configuration. Une exigence peut être associée à un élément dont la configuration est gérée.
- Test – Le Test fournit des informations sur les différences entre les exigences exposées et les caractéristiques fonctionnelles/non fonctionnelles du produit développé. La traçabilité entre les exigences, tests et autres objets est obligatoire afin de fournir une visibilité complète au regard de la couverture de test. Selon les éléments de test associés, une exigence spécifique peut avoir un état indiquant si oui ou non elle est mise en œuvre.
- Gestion de version - Le point de départ de la gestion des versions est le résultat de la gestion des exigences. Pour chaque solution ou produit versionné, il doit être possible d'identifier les exigences mises en œuvre dans cette version.
Avec différents modèles de développements, ou différents produits, l'Ingénierie des Exigences peut être ainsi reliée à d'autres processus.

[Pour les organismes de formation : fournir des exemples de relations entre l'Ingénierie des Exigences et d'autres disciplines.]

3 Processus d'Ingénierie des Exigences	80 minutes
---	-------------------

Termes

Client, Processus d'Ingénierie des Exigences, développeur des exigences, Ingénieur des exigences, gestionnaire des exigences, fournisseur, parties-prenantes

Objectifs d'apprentissage pour le niveau fondation de l'Ingénierie des Exigences

Les objectifs identifient ce que vous pourrez démontrer à l'issue de chaque module.

3.2 Processus Générique d'Ingénierie des Exigences

- LO-3.2.1 Expliquer les principaux objectifs de l'Ingénierie des Exigences et les activités typiques du processus d'Ingénierie des Exigences (K2)

3.3 Rôles et Responsabilités

- LO-3.3.1 Expliquer les rôles du Développement des Exigences et de la Gestion des Exigences (K2)
- LO-3.3.2 Se souvenir des principaux rôles dans l'Ingénierie des Exigences et expliquer leur action dans le processus d'Ingénierie des Exigences (K1)
- LO-3.3.3 Se souvenir des principaux talents et compétences requis d'une personne impliquée dans l'Ingénierie des Exigences (K1)

3.1 Introduction au processus d'Ingénierie des Exigences

40 minutes

La structure du processus d'Ingénierie des Exigences dépend de différents facteurs, tels que la culture et la maturité de l'organisation, le modèle de développement utilisé, etc. REQB utilise la classification suivante :

- Le processus d'Ingénierie des Exigences
- Le processus d'Ingénierie des Exigences dans le développement ainsi que les approches et modèles de maintenance
- Le processus d'Ingénierie des Exigences dans les modèles de maturité.

Le processus d'Ingénierie des Exigences devrait être le point de départ de toute organisation impliquée dans le travail de développement de solution car il fournit les processus les plus cruciaux pour les exigences traitées. Ce processus d'Ingénierie des Exigences peut être adapté aux besoins spécifiques de l'organisation et le modèle de développement ou de maintenance qui doit être appliqué. Le résultat de cette adaptation est le processus d'Ingénierie des Exigences dans les modèles et les approches de développement et de maintenance. En plus des modèles de processus de développement, il existe des modèles de maturité. Ces modèles de maturité sont principalement utilisés pour évaluer la maturité actuelle d'une organisation spécifique et apporter des améliorations nécessaires pour accroître l'efficacité de l'organisation. Par conséquent, nous pouvons définir une autre variante du processus d'Ingénierie des Exigences – le processus d'Ingénierie des Exigences dans les modèles de maturité. Cette variante du processus d'Ingénierie des Exigences est couverte dans le syllabus REQB niveau avancé.

Le processus d'Ingénierie des Exigences doit être adapté aux besoins spécifiques de l'organisation, en tenant compte du type de solutions en cours de développement, du domaine métier, des processus métier, de la culture de l'organisation, du niveau de compétence et de la capacité du personnel en charge du processus de l'Ingénierie des Exigences.

[Pour les organismes de formation : expliquer l'approche REQB dans le processus d'Ingénierie des Exigences et démontrer la nécessité d'ajuster le processus d'Ingénierie des Exigences pour des besoins spécifiques.]

3.2 Processus Générique d'Ingénierie des Exigences	20 minutes
---	-------------------

L'Ingénierie des Exigences est une discipline qui inclut les processus nécessaires pour l'identification, la structuration et la gestion des Exigences. Les activités spécifiques couvertes par le processus d'Ingénierie des Exigences sont les suivantes :

- Identification des exigences
- Analyse des exigences
- Spécification des exigences
- Vérification et validation des exigences
- Traçabilité des exigences
- Gestion de Configuration et Gestion de changement
- Assurance Qualité

Le processus d'Ingénierie des Exigences est un ensemble structuré des activités mentionnées ci-dessus. Les activités sont organisées dans le cadre d'un processus de gestion des exigences ou un processus de développement des Exigences, selon le but et la phase de l'élaboration de la solution. Une description complète du processus devrait inclure des connexions avec d'autres disciplines ainsi que les domaines connexes (par exemple, analyse métier ou Tests), l'entrée et la sortie des activités, des informations sur la responsabilité pour une activité spécifique et ses résultats, les compétences requises et les outils supportant ces activités (plus de détails dans le chapitre 7 « Outils »).

Les principales activités, entrées et sorties des processus d'Ingénierie des Exigences sont présentées dans le tableau 2.

Processus	Entrées	Principales activités	Sorties
Développement des exigences			
<ul style="list-style-type: none"> • Identification des exigences 	<ul style="list-style-type: none"> • Besoins métier • Définition du périmètre initial • Parties-prenantes 	<ul style="list-style-type: none"> • Collecter les exigences en provenance des parties-prenantes • Détailler les exigences haut-niveau • Exclure les caractéristiques non nécessaires 	<ul style="list-style-type: none"> • Exigences métier • Limites • Hypothèses
<ul style="list-style-type: none"> • Analyse des exigences 	<ul style="list-style-type: none"> • Exigences métier • Limites • Hypothèses • Parties-prenantes • Ressources de l'organisation • Techniques de modélisation • Outils 	<ul style="list-style-type: none"> • Transformer les exigences métier en exigences solution/système et les affiner en exigences composants/produit • Résoudre les conflits (exigences contradictoires) • Etablir les frontières de la solution • Développer les modèles de la solution métier 	<ul style="list-style-type: none"> • Accord sur les exigences • Périmètre de la solution défini • Modèles de solution métier
<ul style="list-style-type: none"> • Spécification des exigences 	<ul style="list-style-type: none"> • Exigences • Limites • Hypothèses • Parties-prenantes • Ressources de l'organisation 	<ul style="list-style-type: none"> • Exigences documentées 	<ul style="list-style-type: none"> • Documentation de spécification d'exigences

Processus	Entrées	Principales activités	Sorties
<ul style="list-style-type: none"> Vérification et validation des exigences 	<ul style="list-style-type: none"> Exigences Documentation de spécification d'exigences 	<ul style="list-style-type: none"> Vérifier la qualité des exigences et la documentation de spécification des exigences 	<ul style="list-style-type: none"> Exigences validées Documentation de spécification d'exigences validée Défauts Résultats Changements
Gestion des exigences			
<ul style="list-style-type: none"> Traçabilité des exigences 	<ul style="list-style-type: none"> Exigences Parties-prenantes Ressources de l'organisation Outils 	<ul style="list-style-type: none"> Etablir une structure pour la traçabilité Définir et maintenir la traçabilité des exigences 	<ul style="list-style-type: none"> Matrice de traçabilité
<ul style="list-style-type: none"> Gestion de la configuration et du changement 	<ul style="list-style-type: none"> Exigences Outils Demandes de changements Plans projet Matrice de traçabilité 	<ul style="list-style-type: none"> Identifier et gérer les éléments de configuration Etablir les demandes, la faisabilité, la planification, la mise en œuvre et l'évaluation des changements à un système, des documents ou d'autres produits du projet Traitement de l'analyse des risques Gérer les changements et assurer la traçabilité des changements 	<ul style="list-style-type: none"> Référentiel d'exigences Résultats de l'analyse des risques Exigences modifiées et document de spécification des exigences Plans projet

Processus	Entrées	Principales activités	Sorties
<ul style="list-style-type: none"> Assurance qualité 	<ul style="list-style-type: none"> Exigences Document de spécification des exigences Modèles de solution Parties prenantes Ressources de l'organisation Standards Outils Matrice de traçabilité Analyse des risques 	<ul style="list-style-type: none"> S'assurer que les différents processus d'Ingénierie des Exigences et leurs produits sont de bonne qualité Appliquer les standards appropriés Gérer les revues et autres activités QA 	<ul style="list-style-type: none"> Exigences Document de spécification des exigences Modèles de solution métier Défauts Résultats Changements Améliorations Processus de l'Ingénierie des Exigences et activités définies et qualifiées

Table 1 - Vue d'ensemble des activités de l'Ingénierie des Exigences, entrées et sorties

L'efficacité du processus d'Ingénierie des Exigences n'est pas la même dans toutes les organisations ou projets. Il existe certains facteurs, internes ou externes, qui peuvent avoir une influence négative sur le processus ou produits de l'Ingénierie des Exigences.

Les facteurs d'un point de vue interne (au sein de l'organisation du fournisseur de logiciels) :

- Manque de connaissances du domaine métier ou utilisateur
- Manque de connaissances des technologies demandées par le client
- Inefficacité de l'approche Méthodologie/Outils de l'Ingénierie des Exigences
- Insuffisance d'expérience et de compétence du personnel

Les facteurs d'un point de vue externe (en dehors de l'organisation du fournisseur de logiciels) :

- Manque de communication ou approche inefficace à la communication des exigences, besoins ou attentes

- Manque de clarté et/ou changement des objectifs métiers résultant dans des exigences instables
- Insuffisance de connaissance sur le processus de développement de produits
- Manque d'implication des utilisateurs et/ou des parties prenantes métier

L'Ingénierie des Exigences traite de nombreuses tâches, selon son rôle dans le développement de la solution, le cycle de vie et de maintenance ainsi que selon les contraintes et les caractéristiques de la solution envisagée.

Le processus d'Ingénierie des Exigences et de ses produits (tels que les spécifications des exigences ou une proposition de solution métier) devrait offrir une valeur ajoutée pour le client et les autres parties prenantes. Pour être en mesure de s'assurer que la solution envisagée répond le mieux aux besoins du client et aux attentes rencontrées, les exigences devraient être développées à partir du point de vue du client.

Les méthodes de processus d'Ingénierie des Exigences orientées client sont les suivantes :

- Analyse et conception orientées utilisateur
- Approche par prototypage
- Utilisation de démonstrations comme moyen de validation des incréments

[Pour les organismes de formation : fournir une vue d'ensemble des activités d'Ingénierie des Exigences, les entrées et sorties d'un domaine métier spécifique/projet.]

3.3 Rôles et Responsabilités

20 minutes

Un processus d'Ingénierie des Exigences réussi nécessite de clarifier les rôles et de définir les responsabilités. En général, les rôles peuvent être classés comme des rôles qui affectent ou sont affectés par l'Ingénierie des Exigences et des rôles au sein de l'Ingénierie des Exigences.

Les rôles les plus importants affectant ou affectés par l'Ingénierie des Exigences sont le client et le fournisseur.

Le client est une personne, un groupe ou une organisation exigeant une solution, tandis que le fournisseur (également appelé un vendeur) est une personne, un groupe ou une organisation fournissant la solution. Le client formule ses besoins et fournit les besoins et les attentes métier initiales. Elles sont généralement fournies en même temps que la demande d'offre / service.

Le fournisseur est responsable de la compréhension de ces besoins et de l'extraction des exigences à partir de ces données. Le principal objectif du fournisseur est de fournir des solutions pour répondre aux besoins du client.

Les rôles au sein de l'Ingénierie des Exigences peuvent être définis comme étant le gestionnaire des exigences et le développeur des exigences. Les deux rôles peuvent aussi être nommés « Ingénieur des exigences ».

La grande variété de tâches liées à l'Ingénierie des Exigences a entraîné de nombreux titres différents pour les personnes affectées à des tâches d'Ingénierie des Exigences. Différentes organisations utilisent leurs propres dénominations tels que « Ingénieur exigences », « Gestionnaire d'Exigences », « Développeur d'Exigences », « Analyste Métier », « Analyste Système », « Architecte Solution », « Architecte Système », « Concepteur », etc.

Les variantes régionales de ces rôles existent en fonction de la culture, des habitudes et des traditions. Le problème est que les responsabilités et les compétences nécessaires concernant les rôles spécifiques mentionnés ci-dessus sont rarement définis, clairs ou compréhensibles. Afin de fournir une compréhension commune des rôles et responsabilités, REQB définit les rôles de gestionnaire et développeur des exigences. Cette classification résulte de la différence dans les tâches accomplies propres aux activités de développement et de gestion des exigences.

Un gestionnaire d'exigences est une personne responsable de la documentation, l'analyse, la traçabilité, la priorisation et l'accord sur les exigences mais aussi le contrôle des changements et la communication des exigences vers les parties prenantes correspondantes.

Un développeur d'exigences est une personne principalement impliquée dans l'élicitation, l'analyse, la documentation et la priorisation des exigences.

Un gestionnaire d'exigences et un développeur d'exigences doivent posséder les compétences suivantes :

- Compétence méthodologique (c'est à dire, une connaissance pratique du processus d'Ingénierie des Exigences, des méthodes, techniques et outils)
- La connaissance des sources de bonnes pratiques et les normes les plus importantes liées à l'Ingénierie des Exigences
- Pensée précise, analytique et claire
- Capacité à agir de manière structurée
- Compétence de modération et de négociation
- Confiance en soi
- Capacité à argumenter et convaincre
- Compétences linguistiques et de communication
- Résistance au stress

Un autre rôle associé à l'Ingénierie des Exigences est une partie prenante. Une partie prenante est un groupe ou un individu qui est affecté par ou est en quelque sorte responsable du résultat d'une action. Les parties prenantes du projet sont des individus et des organisations qui sont activement impliqués dans le projet ou dont les intérêts peuvent être affectés à la suite de l'exécution du projet ou de l'achèvement du projet.

Les parties prenantes peuvent être des personnes physiques, des sociétés ou des personnes morales.

Les parties prenantes ont souvent des conflits d'intérêts entre elles. Cela aboutit souvent à des exigences contradictoires. Le problème des exigences contradictoires doit être abordé lors de l'activité d'analyse des exigences.

Il peut y avoir plusieurs catégories de parties prenantes, parmi lesquelles :

- Les clients
- Les utilisateurs finaux
- Les managers (Chefs de projet)
- Les Ingénieurs chargés du développement et de la maintenance du système
- Les clients de l'organisation qui utilisera le système
- Organismes externes (par exemple, les organismes de réglementation)
- Les experts du domaine
- Fournisseurs de solutions liées
- Les organismes de maintenance

L'identification de tous les acteurs est nécessaire pour prendre en considération tous les points de vue dans la solution envisagée. Dans le cas de parties-prenantes manquantes ceci peut entraîner des exigences oubliées ou des limitations importantes influant sur la portée ou la forme de la solution.

[Pour les organismes de formation : décrire un acteur typique, ses points de vue sur le projet et les livrables, les principaux objectifs et les modes de communication (par exemple, propriétaire/sponsor métier, gestionnaire de projet, client, Ingénieur d'exigences, équipe de développement Agile, Scrum Master).]

4 Gestion des Exigences	160 minutes
--------------------------------	--------------------

Termes

Référentiel, changement, comité de contrôle du changement, gestion du changement, demande de changement, gestion de configuration, élément de configuration, traçabilité horizontale, analyse d'impact, métriques, assurance qualité, contrôle qualité, risques produits, risques projet, revues, risque, analyse de risque, identification des risque, gestion des risques, plan de gestion des risques, atténuation du risque, traçabilité, traçabilité verticale.

Objectifs d'apprentissage pour le niveau fondation de l'Ingénierie des Exigences

Les objectifs identifient ce que vous pourrez démontrer à l'issue de chaque module.

4.2 Gestion de Projet et Gestion du Risque

- LO-4.2.1 Expliquer pourquoi l'Ingénierie des Exigences est importante dans les projets et fournir des exemples appropriés (K2)
- LO-4.2.2 Se souvenir des erreurs qui peuvent survenir en Ingénierie des Exigences (K1)
- LO-4.2.3 Faire la distinction entre des risques projet et des risques produit en lien avec les exigences (K2)
- LO-4.2.4 Décrire, à l'aide d'exemples, comment l'Analyse des Risques et la Gestion des Risques peuvent être utilisées pour la Gestion des Exigences (y compris la planification des exigences) (K2)

4.3 Traçabilité des Exigences

- LO-4.3.1 Comprendre le but de la traçabilité (K2)
- LO-4.3.2 Reconnaître les différents types de traçabilité (K1)

4.4 Gestion de Configuration et Gestion du Changement

- LO-4.4.1 Expliquer les caractéristiques de la Gestion de Configuration, de la Gestion du Changement ainsi que le rôle d'un Comité de Contrôle du Changement (K2)
- LO-4.4.2 Expliquer les termes : élément de configuration, défaut et demande de changement (K2)

4.5 Assurance Qualité

- LO-4.5.1 Se souvenir des facteurs qui influent sur la qualité du processus et des produits de l'Ingénierie des Exigences (K1)
- LO-4.5.2 Expliquer comment les produits d'activité de l'Ingénierie des Exigences contribuent au test (K2)
- LO-4.5.3 Expliquer comment des métriques peuvent être utilisées pour évaluer et améliorer la qualité du processus d'Ingénierie des Exigences et ses livrables (K2)

4.1 Introduction à la Gestion des Exigences

10 minutes

La Gestion des exigences est essentiellement une série d'activités définies pour s'assurer que le processus de développement des Exigences est mis en œuvre correctement pendant le cycle de vie du produit. La gestion de configuration, la gestion du changement ainsi que la maintenance de la solution sont également des activités de la gestion des exigences.

Généralement la gestion des exigences inclut les activités suivantes :

- Traçabilité des exigences
- Gestion de Configuration et Gestion du changement
- Assurance Qualité

4.2 Gestion de Projet et Gestion du Risque

40 minutes

4.2.1 Introduction

Certaines des principales raisons de l'échec de projets sont liées aux exigences. La négligence vis-à-vis de l'Ingénierie des Exigences peut entraîner des exigences imprécises, contradictoires, ne répondant pas aux critères ou ne satisfaisant pas les besoins des parties prenantes. C'est pourquoi, une Ingénierie des Exigences structurée et précautionneuse est nécessaire à chaque projet.

Afin de minimiser les risques projet, l'Ingénierie des Exigences devrait contribuer aux domaines suivants :

- Cadrage projet
 - Identification des besoins et des attentes du client, des parties-prenantes, objectifs et vision concernant la solution du problème
 - Définition des exigences de haut niveau
- Négociations de contrat
 - Évaluation des exigences client
 - Détermination du périmètre initial et des ressources requises pour un projet
 - Détermination du coût de développement (implémentation des exigences)
- Définition du projet
 - Définition des rôles, des tâches, des activités et des processus additionnels (exemple : gestion du changement)
 - Contribution à la conception de l'architecture initiale
 - Contribution aux livrables du processus de test
- Réalisation projet
 - Fournir une base pour le développement, la vérification et validation des Exigences
 - Contrôler les changements, forcer la revue des plans et leur ajustement au périmètre courant de la solution en cas de changement des exigences

4.2.2 Problèmes habituels dans la Gestion des Exigences

L'inefficacité ou l'insuffisance des exigences augmente les risques du projet en permettant à des erreurs de se propager à des étapes ultérieures dans le développement de solutions. Par conséquent, lors de la planification du processus d'Ingénierie des Exigences, les meilleures pratiques, les risques ainsi que les erreurs courantes doivent être pris en compte.

Les problèmes les plus courants liés à l'Ingénierie des Exigences sont les suivants :

- Le manque de clarté des exigences métier ainsi que le manque d'objectifs métier pour la réalisation de la solution
- Les changements d'exigences résultent souvent d'objectifs projet non clairs ainsi que de l'absence d'une définition claire du domaine métier du client. Les changements des exigences ne sont pas perçus comme une erreur dans les approches itératives et agiles.
- Les responsabilités mal définies (à la fois coté client et fournisseur)
- Les écarts entre les attentes des parties prenantes et les livrables du projet
- L'implication insuffisante du client
- Le manque de traçabilité, ce qui entraîne souvent des estimations imprécises de l'impact des changements d'exigences sur les autres parties du produit en cours de développement
- Estimation imprécise des coûts ou du périmètre, et / ou de la définition des jalons du projet qui ne peuvent être atteints (souvent résultant d'exigences non claires).

4.2.3 Gestion des Risques

Les problèmes potentiels énumérés ci-dessus peuvent être perçus comme un risque.

Pour faire face à ces risques et d'autres risques potentiels, un processus de gestion des risques est nécessaire.

Une gestion des risques efficace est une clé pour la réduction des risques projet et produit. L'identification, l'analyse appropriée ainsi que la planification des réponses adéquates aux risques aident à minimiser la probabilité qu'un risque se réalise avec les conséquences qui en découlent. Les risques ne sont pas isolés vis-à-vis des exigences. En général, le risque peut être défini comme étant la probabilité qu'un événement ou une situation se produise et entraîne des conséquences indésirables ou un problème potentiel. Le risque est exprimé sur une échelle de niveaux, où le

niveau de risque est déterminé à la fois par la probabilité qu'un événement indésirable se produise ainsi que par son impact (le préjudice résultant de cet événement) [ISTQB].

Il existe deux principaux types de risques : le risque du produit et le risque du projet.

Les risques projet sont les risques qui empêchent le projet d'atteindre ses objectifs, tels que :

- Les facteurs organisationnels (par exemple, les compétences, la formation et le manque de personnel, les questions politiques, les problèmes avec les parties prenantes pour communiquer leurs besoins et attentes au regard de la solution envisagée, une attitude ou attente incorrecte envers l'Ingénierie des Exigences)
- Les problématiques techniques (par exemple, les problèmes dans la définition des bonnes exigences, les solutions technologiques / architecture, la mesure selon laquelle les exigences ne peuvent être satisfaites compte tenu des contraintes existantes, la mauvaise qualité de la conception, du code, des données de configuration, des données de tests et des tests, des spécifications d'exigences ou de la solution)
- Les problématiques du fournisseur (par exemple, des composants développés par un tiers non livrés à temps, ou les problématiques contractuelles)
- Le risque métier basé sur la mauvaise qualité (par exemple, le risque de perdre des clients si la solution fournie est jugée non fiable ou inefficace)

Les risques produit sont des zones d'échec potentielles (événements futurs adverses ou danger) dans le logiciel ou système, ils sont un risque sur la qualité du produit. Cela inclut :

- Risque important d'échec de livraison du logiciel (logiciel ou système incapable de fournir une fonction requise dans les limites spécifiées), documentation de la solution de basse qualité (ex manuel utilisateur, guide d'installation)
- Le potentiel qu'un logiciel pourrait causer comme préjudice à un individu ou une société
- Les caractéristiques qualité du produit médiocres (par exemple, fonctionnalité, fiabilité, facilité d'emploi ou performance)
- La mauvaise qualité et intégrité des données (par exemple, problématiques de migration de données, problèmes de conversion de données, problèmes de transport de données, violation de standards de données)
- Logiciel qui ne réalise pas les fonctionnalités attendues et qui ne satisfait pas les besoins des parties-prenantes

Le processus de gestion du risque permet d'identifier les facteurs potentiels qui peuvent avoir un effet négatif sur l'exécution d'un projet et préparer une action appropriée pour faire face à un risque s'il survient.

La gestion du risque comprend les activités suivantes [ISTQB] :

- L'identification du risque
- L'analyse du risque
- L'atténuation du risque

Niveau fondation

Lors de l'identification des risques, il est important d'interroger l'ensemble des parties prenantes, étant donné que les différents groupes de parties prenantes ou acteurs individuels peuvent reconnaître les différents risques. Par exemple, un sponsor métier peut être concerné par la valeur fournie par la solution, tandis que le risque le plus important pour un ingénieur des exigences peut-être un manque de communication avec les représentants métier du client.

L'analyse des risques consiste à établir les niveaux de risque - la probabilité et l'impact potentiel. Les deux attributs peuvent être exprimés selon des niveaux, ou - plus précisément – sous la forme d'estimations numériques (par exemple, la probabilité d'un événement indésirable se passe peut être exprimée comme une valeur de pourcentage, tandis que l'impact peut être exprimé comme une perte financière). L'atténuation des risques prévoit des réactions appropriées portant sur les risques identifiés basées sur les résultats de l'analyse des risques.

Les techniques de base pour atténuer les risques peuvent être divisées en 4 catégories principales :

- Evitement
- Réduction
- Partage
- Maintien

Lorsque vous travaillez sur le processus de gestion des risques, il est important d'établir et de maintenir un plan de gestion des risques. Ce plan doit être créé avant la création du plan de projet et mis à jour périodiquement (par exemple, après chaque itération ou étape). Le plan de gestion des risques devrait fournir des contrôles de sécurité efficaces pour gérer les risques et doit contenir un planning pour le contrôle de l'implémentation et une liste des personnes responsables de ces actions.

Le plan de gestion des risques comprend généralement :

- Liste des risques (classés en types)
- Le propriétaire
- La probabilité d'occurrence de chaque risque
- La sévérité de l'impact de chaque risque (y compris les coûts, le cas échéant)
- Les stratégies d'atténuation pour chacun des risques (incluant la personne / groupe responsable de prendre les actions d'atténuation des risques)
- La traçabilité des risques aux exigences et autres artefacts de projet
- La matrice d'évaluation des risques

[Pour les organismes de formation : fournir des exemples de risques liés aux exigences et des exemples d'actions pour gérer les risques.]

4.3 Traçabilité des Exigences

40 minutes

4.3.1 Instabilité des Exigences

Les exigences ne sont jamais figées et continuent à évoluer pendant le cycle de vie de projet. Les raisons de cette évolution continue et des changements induits sont notamment issues :

- De nouvelles analyses ou nouveaux besoins des clients (par exemple, intégrer une nouvelle réglementation, de nouvelles évolutions métier et/ou commerciales, de produits nouveaux)
- D'un travail continu (par exemple : détailler les exigences de haut niveau déjà définies, anticiper la prochaine phase de projet, améliorer et optimiser les caractéristiques déjà mises en application)
- De nouvelles interfaces pour le projet (par exemple : intégrer de nouveaux systèmes, de nouveaux canaux de communication tels que l'Internet ou les mobiles)

4.3.2 Cycle de Vie des Exigences

Selon le degré d'analyse des besoins et/ou de leur mise en œuvre, un état ou statut particulier est affecté à chaque exigence.

Ce statut est géré par la notion de cycle de vie d'une exigence, et est souvent exprimé par l'intermédiaire de différents statuts, par exemple :

- Nouveau (proposé)
- Approuvé
- En conflit
- Mis en œuvre
- Modifié
- Supprimé

- Déployé

Les approches et organisations étant différentes suivant les contextes, elles peuvent utiliser différents cycles de vie des exigences (et différents statuts). Dans de nombreux cas, les cycles de vie des exigences, les demandes de changement et les défauts sont très semblables et gérés en utilisant le même outil.

4.3.3 Traçabilité des Exigences

La traçabilité fournit une méthode pour gérer les évolutions des exigences ainsi que celles des artefacts liés à ces exigences.

La traçabilité fournit un moyen de contrôle pour s'assurer que les étapes importantes du processus de développement des exigences ont été effectuées. La traçabilité devrait être bidirectionnelle pour tous les artefacts (par exemple : des exigences vers ses artefacts de conception et des artefacts de conception vers les exigences correspondantes en amont). La traçabilité sert également à la détermination de la portée de la base de test, à la vérification et à la validation.

Les principaux objectifs de la traçabilité sont les suivants :

- Analyse d'impact
- Analyse de couverture
- Preuve de mise en œuvre
- Utilisation de l'exigence (analyse des liens entre exigences pour fournir la preuve que les exigences sont utilisées et comment elles sont utilisées)
- Réutilisabilité des exigences

Afin d'assurer une bonne traçabilité, il est important d'affecter une nomenclature unique aux exigences.

Il y a deux types de base de traçabilité :

- Traçabilité horizontale
 - Présenter les dépendances entre différents types d'artefact au même niveau d'abstraction

- Traçabilité verticale
 - Présenter les dépendances entre les artefacts à différents niveaux d'abstraction

Dans le cadre des exigences, la traçabilité horizontale est habituellement employée pour présenter les relations entre les différents types d'exigences (par exemple : traçabilité horizontale entre une exigence fonctionnelle de logiciel et une exigence IHM correspondante). La traçabilité verticale est souvent employée pour contrôler la couverture des exigences à des niveaux différents d'abstraction (par exemple : traçabilité verticale entre les exigences métier et les exigences de solution).

Les deux types de traçabilité peuvent être employés pour réaliser les analyses d'impacts.

Selon la norme ISO / IEC / IEEE 29148 : 2011 (auparavant IEEE 830), les modèles statiques de traçabilité peuvent être mis en œuvre par des :

- Références textuelles dans les exigences reliées aux identifiants des autres artefacts
- Références textuelles dans les attributs
- Hyperliens
- Matrices

[Pour les organismes de formation : expliquer le concept de la traçabilité et fournir les exemples de son application.]

4.4 Gestion de Configuration et Gestion du Changement	30 minutes
--	-------------------

4.4.1 Introduction

Pour assurer une bonne gestion des exigences, un processus de gestion de configuration doit être mis en application. Dans de nombreux cas les exigences ne sont pas stables et les changements induits peuvent affecter les autres artefacts du projet.

4.4.2 Gestion de Configuration

Le but de la gestion de configuration est d'établir et mettre à jour l'intégrité des produits (composants, données, et documentation) et des artefacts du logiciel, et ceci durant tout le cycle de vie du projet ainsi que le cycle de vie du produit.

La gestion de configuration est une discipline qui nécessite des outils techniques et d'administration pour identifier et documenter les caractéristiques fonctionnelles et physiques d'un élément de configuration, contrôler les changements des caractéristiques, enregistrer et fournir le dossier de changement et, enfin, vérifier la conformité avec les exigences spécifiées [IEEE 610]

Un élément de configuration est un artefact, document, produit (matériel et/ou logiciel) qui a un but pour l'utilisateur final et qui est traité comme un élément unique et simple dans le processus de gestion de configuration [IEEE 610].

Les principaux éléments de configuration utilisés dans l'Ingénierie des Exigences incluent :

- Les exigences en tant qu'élément individuel
- La spécification des exigences
- Les modèles

La gestion de la configuration s'assure que tous les résultats de l'Ingénierie des Exigences sont identifiés, que les versions sont contrôlées, que les changements sont tracés, que les exigences sont reliées entre elles et reliées aux autres éléments du projet (des artefacts par exemple de dé-

veloppement et de test) afin que la traçabilité puisse être mise à jour dans tout le processus de fabrication.

Lors de la planification du projet, les procédures de gestion de configuration et de l'infrastructure (outils) doivent être choisies, documentées et mises en œuvre. Ceci est nécessaire dans la mesure où les éléments de configuration doivent être définis et mis sous le contrôle du changement dès le démarrage du cycle de vie du projet.

[Pour les organismes de formation : fournir des exemples des activités de gestion de configuration pour différents résultats du processus d'Ingénierie des Exigences.]

4.4.3 Gestion du Changement

Un aspect important lié à la configuration est le changement. Des changements d'exigences peuvent être demandés à tout moment pendant la réalisation du projet et impacter le périmètre du produit final.

Une source de changement peut être :

- Extension ou changement de la fonctionnalité existante
- Demande de nouvelle fonctionnalité ou exigence
- Défauts trouvés dans le produit/documentation nécessitant un besoin de changement
- Changements résultant de facteurs externes (par exemple : changements organisationnels, changements de réglementation)

Les changements se produisent constamment et il est important de les prévoir en termes de processus et calendrier.

Il est important de distinguer le changement du défaut. Un défaut est une déviation de l'état exigé du système. Un changement est une évolution ou un ajout de caractéristiques, d'exigences ou de fonctionnalités.

Les changements sont gérés par le processus de gestion du changement. Le processus de gestion du changement est le processus qui permet la demande de changement, la détermination de l'opportunité, la planification, la mise en œuvre, et les impacts liés au logiciel/au système, aux

documents ou à d'autres produits du projet. Le but de la gestion du changement est de supporter le traitement des changements et d'assurer la traçabilité des changements.

Le processus de gestion du changement inclut normalement les activités suivantes :

1. Identification d'un changement potentiel
2. Demande nouvelle/modification d'exigence (soumission d'une demande de changement)
3. Analyse de la demande de changement (habituellement faite par le comité de contrôle des changements et en incluant l'analyse d'impact)
4. Évaluation du changement (évaluation y compris du risque, du coût, et de l'effort de la mise en place du changement afin de prendre une décision) → à ce niveau il y a un "GO" "NOGO"
5. Planification du changement (si le changement a été approuvé pour la mise en œuvre)
6. Mise en œuvre du changement
7. Revue et clôture du changement
8. Mise en place du changement dans le périmètre du projet

4.4.4 Comité de Contrôle des Changements

Un changement devrait être soumis formellement sous la forme d'un document de demande de changement (parfois appelé une demande de changement, RFC – Request For Change pour l'acronyme anglais). Ce document comporte à minima les informations suivantes :

- Le nom de la personne/du service ou de toute autre entité demandant le changement
- Date de demande
- La raison du changement
- Une description de la solution demandée
- Date prévue (ou désirée) de la mise en place du changement (si c'est approprié)
- Budget prévu (ou autorisé) pour le changement (si c'est approprié)

Des changements sont habituellement contrôlés et décidés par le comité de contrôle des changements ou le comité de configuration (acronyme anglais CCB – Change Control Board). Le comité de

contrôle des changements prend les décisions au regard des changements proposés en se basant sur les informations fournies (telles que le risque lié au changement, à son incidence, et l'effort requis pour la mise en place). Le CCB se compose de parties prenantes de projet ou de leurs représentants. Dans les projets agiles, le rôle du CCB est habituellement exécuté par l'équipe de développement ainsi que par les représentants du client et le propriétaire du produit (product owner).

Mettre en place un comité de changement (CCB) nécessite des procédures pour passer en revue, évaluer et mettre en place un changement à un produit ou à un service.

Le comité de contrôle des changements (CCB) comprend couramment les rôles suivants :

- Gestion de projet
- Ingénierie des Exigences
- Gestion du développement (au sens technique)
- Assurance Qualité (gestion de la qualité, gestion des Tests)
- Gestion Métier
- Représentants du client et/ou d'utilisateur final

Selon sa complexité et ses impacts, un changement peut induire divers effets sur le système prévu. Les petits changements peuvent induire des changements mineurs, ou un changement complexe peut radicalement changer la logique du système. Chaque changement devrait être soigneusement analysé afin d'identifier les risques potentiels associés et évaluer la valeur de ce changement en tenant compte des risques prévus.

En décidant d'un changement il est important d'analyser l'impact de ce changement sur le projet. Ceci est habituellement fait par l'analyse d'impact. L'analyse d'impact utilise la traçabilité pour identifier quels artefacts devront être changés ou au moins devraient être contrôlés en introduisant le changement.

Les changements des exigences peuvent induire divers effets sur le projet. Les effets les plus courants sont (selon la phase du projet) :

- Mise à jour des artefacts d'analyse et de conception (par exemple, exigences ou caractéristiques de solution)
- Mise à jour de la documentation technique et d'utilisation
- Evolution de la stratégie de test, des plans de test et des cas de test
- Mise à jour du plan de formation (voir le besoin)
- Augmentation ou diminution du périmètre de développement du système

- Evolution de la portée des Tests
- Modifications de la planification, du budget, et des ressources

Des changements mis en application devraient être vérifiés avant d'être déployés dans les environnements de développement système et de Test.

4.5 Assurance Qualité

40 minutes

4.5.1 Introduction

En définissant le processus de gestion des exigences, il est également nécessaire de définir les activités d'Assurance Qualité (QA) à appliquer afin de s'assurer que les différents processus d'Ingénierie des Exigences et leurs produits sont de bonne qualité.

L'assurance Qualité est définie comme « toutes les activités prévues et systématiques décrites dans le système qualité, et sélectionnées comme nécessaires, pour fournir la confiance adéquate qu'une entité requiert globalement » [ISO 9000]. Cette définition implique que les mesures prises sont « prévues et systématiques » et qu'elles « fournissent la confiance adéquate » c'est à dire que le niveau de qualité désiré sera atteint. Ces actions incluent des techniques et activités opérationnelles pour atteindre complètement les critères définis pour la qualité.

Pour atteindre le niveau requis de qualité, le contrôle qualité est aussi nécessaire. L'objectif principal du contrôle qualité est d'orienter et contrôler la qualité des produits ou des services par l'utilisation de méthodes spécifiques de sorte qu'elles répondent aux normes spécifiques de qualité. Ces méthodes spécifiques au regard de l'Ingénierie des Exigences incluent la gestion de projet, la gestion des risques, la gestion du changement, la vérification et la validation, les revues, la gestion de la configuration et la gestion de la traçabilité des exigences.

4.5.2 Facteurs influençant le niveau de qualité attendu

Dans le cadre de l'Ingénierie des Exigences, le contrôle qualité peut se concentrer sur la vérification en s'assurant que la documentation produite pour les exigences répond aux critères de qualité souhaités.

En planifiant les activités d'assurance qualité pour un projet, il est important de se rappeler qu'il y a un certain nombre de facteurs qui peuvent affecter le niveau de qualité désiré du produit. De tels facteurs peuvent inclure :

- Les critères ou les attentes qualité du client vérifiables par le processus d'assurance qualité

- Le type de produit qui est implémenté (par exemple : la complexité versus le public cible du produit - un produit destiné à une petite assistance pourrait avoir un niveau moins exigeant de qualité qu'un produit grand public)
- L'environnement dans lequel le produit est implémenté (par exemple : l'environnement technique pourrait limiter la capacité d'atteindre le niveau requis de qualité)
- Domaine (par exemple : complexité du domaine métier, le niveau de l'innovation, la fréquence d'évolution du métier)
- Juridique, sécurité, contexte (par exemple : contrainte réglementaire obligeant à atteindre un haut niveau de la qualité)
- Les pressions de temps et coût qui réduisent la capacité d'exécuter correctement l'Ingénierie des Exigences

Afin d'assurer le niveau requis de la qualité, la vérification et la validation devraient être planifiées et exécutées à partir du début du projet et tout au long du projet (Voir section 5.5 validation et vérification des exigences).

Les techniques et outils suivants peuvent être utilisés pour l'assurance qualité et le contrôle qualité des exigences :

- Normes et standards
- Traçabilité
- Différents types de revue
- Prototypage
- Observance des exigences/des critères qualité de cahier des charges

4.5.3 Mesure de la qualité des exigences

L'assurance qualité peut prendre en compte la vérification de la testabilité des exigences. Des exigences testables (vérifiables) peuvent être contrôlées si elles sont mises en application correctement et acceptées par les parties prenantes. La testabilité des exigences est déterminée par les critères d'acceptation du projet (voir la section 5.5 validation et vérification des exigences).

L'assurance qualité des exigences et de la spécification de la solution peut être améliorée en incluant les éléments suivants :

- Le but du document, sa portée, les définitions et le glossaire
- Les objectifs pour les différents niveaux (par exemple : une spécification d'exigences de haut niveau a des objectifs différents de ceux relatifs à une spécification d'exigences fonctionnelles détaillées)
- Contraintes de conception et de réalisation
- Priorités ou catégories pour les exigences
- Définitions claires sur ce que le système devrait faire plutôt que comment il devrait le faire
- Documentation pour toutes législations, suppositions, règles métier et dépendances applicables
- Pas de description supplémentaire inutile des diagrammes qui sont clairs et autonomes (remplacer si possible le texte et les abstractions par des diagrammes)
- Catalogue spécifique des notions utilisateur et plan d'habilitation (droits et privilèges d'utilisateurs)
- Présentation structurée
- Langage simple, clair, précis et non ambigu

Comme stipulé dans le processus générique d'Ingénierie des Exigences de la section 3.2, les exigences sont l'information de base d'entrée des processus de développement et de test. Les exigences bien définies réduisent le risque de projet, et en particulier les dysfonctionnements du futur produit. Les exigences stabilisées augmentent les chances de respecter les échéances d'un projet.

L'Ingénierie des Exigences est étroitement liée au test. Des bons cas de tests exigent des exigences d'un bon niveau de testabilité. La participation des testeurs pendant la construction des spécifications d'exigences et de solution est donc très importante. Les résultats de l'Ingénierie des Exigences supportent le test en constituant la base pour le test. Les exigences et leurs caractéristiques sont désignées parfois sous le nom de base de test (voir ISTQB FL).

Il est important de se rappeler que les exigences devraient être validées par des tests statiques (revues) et peuvent impliquer des testeurs (notamment en mode Agile). L'acceptation par les gestionnaires de test peut être exigée, en particulier quand les exigences se sont avérées d'un faible niveau de testabilité. Les testeurs aident à améliorer la qualité des exigences en identifiant les points faibles et les défauts possibles. Les testeurs devraient également participer aux revues d'exigences afin de s'assurer de leur aptitude au test (niveau de testabilité).

S'assurer et contrôler la qualité exige habituellement des métriques. Une métrique est une échelle de mesure et définit la méthode employée pour la mesure [ISO 14598]. Les métriques permettent

de faire un bilan quantifiable concernant l'état du projet et de sa qualité. Il est important de se rappeler que les résultats d'une mesure (nombres collectés pendant la mesure) doivent toujours être comparés aux données de référence (une métrique appropriée appelée généralement calibration).

Les métriques simples suivantes peuvent être appliquées aux exigences :

- Estimation des coûts
 - Nombre et complexité des exigences
- Stabilité du projet
 - Nombre de changement d'exigences
- Nombre d'erreurs trouvés dans les exigences ou les spécifications
 - Type d'erreurs
 - Nombre d'erreurs avec différents types (par exemple, logique, cohérence, erreurs de données.)

La mesure de la qualité des exigences est souvent difficile car exprimer la qualité d'une abstraction n'est pas simple. En conséquence la qualité est souvent mesurée à l'aide de listes de contrôle couvrant des attributs communs de qualité. Quelques questions à utiliser pour évaluer la qualité des exigences :

- L'exigence est-elle correcte ?
- L'exigence est-elle non ambiguë ?
- L'exigence est-elle faisable ?
- L'exigence est-elle décelable ?
- L'exigence est-elle identifiable ?
- L'exigence est-elle testable ?
- L'exigence est-elle indépendante de la conception ?

Le taux de changement peut également être une mesure pour la qualité des exigences (ceci ne s'applique pas aux projets agiles). Des changements sont apportés pour clarifier des exigences peu claires et pour corriger des exigences inachevées ou contradictoires. Plus le taux de changement de l'ensemble des exigences est élevé, plus le projet est risqué. Par conséquent le taux de changement devrait être mesuré pour contrôler des risques dans un projet.

[Pour les organismes de formation : fournir des exemples de métrique qui mesurent la qualité des exigences et du processus d'Ingénierie des Exigences.]

5 Développement des Exigences	290 minutes
--------------------------------------	--------------------

Termes

Critère d'acceptation, apprentissage, BPMN, brainstorming, modèle conceptuel, contrat, niveaux de formalisation, méthode de Delphi, observation terrain, analyse des points de fonction, but, modélisation IHM, identification sur la base de documents existants, entretiens, modèle, planning poker, priorisation, questionnaire, élicitation des exigences, modèle d'exigences, spécification d'exigences, représentant du client, réutilisation, auto-enregistrement, S.M.A.R.T, modèle de solution, spécification de solution, spécification Sysml, UML, cas d'utilisation analyse des points des cas d'utilisation, user story, vision, atelier

Objectifs d'apprentissage pour le niveau fondation de l'Ingénierie des Exigences

Les objectifs identifient ce que vous pourrez démontrer à l'issue de chaque module.

5.2 Elicitation des Exigences

- LO-5.2.1 Comprendre les objectifs de l'élicitation des exigences (K2)
- LO-5.2.2 Expliquer les sources des exigences et donner quelques exemples (K2)
- LO-5.2.3 Comprendre la signification d'une vision d'un projet dans le contexte de l'Ingénierie des Exigences (K2)
- LO-5.2.4 Comprendre le rôle des parties prenantes pour l'élicitation des exigences et expliquer comment les parties prenantes peuvent être identifiées (K2)
- LO-5.2.5 Expliquer la mise en œuvre des différentes techniques de l'élicitation des exigences et appliquer les pour un scénario donné (K3)

5.3 Analyse des exigences

- LO-5.3.1 Expliquer l'objectif et les activités principales de l'analyse des exigences (K2)

- LO-5.3.2 Expliquer les différences entre le domaine métier et le domaine de la solution (K2)
- LO-5.3.3 Rappeler les raisons de faire des estimations d'effort et fournir des exemples des différentes approches d'estimation (K2)
- LO-5.3.4 Appliquer la procédure de priorisation des exigences pour un scenario donné (K3)
- LO-5.3.5 Expliquer le but et les conséquences de l'acceptation des exigences (K2)
- LO-5.3.6 Rappeler les différents modèles utilisés pendant l'analyse des exigences (K1)
- LO-5.3.7 Comprendre la mise en œuvre et être capable de créer des modèles basiques d'analyse (analyse de contexte, décomposition fonctionnelle, métier) (K3)
- LO-5.3.8 Expliquer les caractéristiques des diagrammes UML (diagramme état-transition, diagramme cas d'utilisation, diagramme d'activités et diagramme de classes) et leur possible mise en œuvre (K2)

5.4 Spécification des exigences

- LO-5.4.1 Comprendre et expliquer l'utilisation des cas d'utilisation et des User Stories dans le développement de la spécification des exigences et de la spécification de la solution (K2)
- LO-5.4.2 Comprendre et expliquer l'objectif, les caractéristiques, le contenu et le mode de création d'une spécification de solution (K2)
- LO-5.4.3 Décrire les caractéristiques principales des différents niveaux de formalisation et expliquer quand utiliser un niveau de formalisation en particulier (K2)

5.5 Vérification et Validation des exigences

- LO-5.5.1 Résumer les techniques de vérification et validation permettant d'éviter les erreurs dans les exigences (K2)
- LO-5.5.2 **Comprendre l'utilisation des critères d'Acceptation (K2)**

5.1 Introduction au Développement des Exigences	10 minutes
--	-------------------

Comme indiqué plus haut, le processus de Développement des Exigences comprend les activités suivantes :

- Elicitation des Exigences
- Analyse des Exigences
- Spécification des Exigences
- Vérification et Validation des exigences

Ces activités ne devraient pas être considérées comme des phases isolées sans interconnexions. Très souvent, la pression du marché pousse à raccourcir le cycle de développement, avec le risque de changements fréquents, de mises à jour ou révisions pour la solution en développement. Par conséquent, il est quasiment impossible de mettre en œuvre de façon linéaire le processus de Développement des Exigences durant lequel les exigences sont collectées auprès des parties prenantes, analysées, spécifiées et transmises à l'équipe de développement qui va ensuite créer les modèles de la solution et valider les exigences. En réalité, les exigences évoluent de façon itérative vers un niveau de détail et de qualité permettant la conception et la prise de décision. Dans la plupart des cas, les exigences seront amenées à changer, ce qui signifie que des mesures devront être prises pour maîtriser les effets des changements.

5.2 Elicitation des Exigences

100 minutes

La première étape du processus de développement des exigences est habituellement l'Elicitation des Exigences. L'objectif principal de l'Elicitation des Exigences est de collecter les exigences de toutes les parties prenantes possibles – pas seulement des utilisateurs ou sponsors, mais de l'équipe projets, du marché ainsi que des autres sources externes. En général, les principaux buts de l'Elicitation des Exigences sont les suivants :

- Identifier toutes les fonctions, caractéristiques, limitations et attentes
- Orienter les exigences vers la vision du projet
- Détailler les exigences de haut niveau et décrire les fonctions et services clairement
- Exclure les fonctions et caractéristiques dont le client ne veut pas

Les sources classiques d'exigences incluent des :

- Documents
- Systèmes utilisés
- Parties prenantes

Ces sources peuvent influencer les techniques retenues pour l'Elicitation des Exigences.

L'Elicitation des Exigences ne consiste pas seulement à collecter les besoins des parties prenantes en posant des questions – très souvent, l'information recueillie doit être interprétée, analysée, modélisée, et validée avant qu'un ensemble complet des exigences d'une solution puisse être établi. Les techniques d'élicitation et les outils à utiliser sont parfois orientés par les approches de modélisation ou d'analyse générale. De nombreuses techniques de modélisation impliquent également l'utilisation de technique d'élicitation particulières.

La première étape de l'Elicitation des Exigences est de préciser le problème à résoudre : identifier les parties prenantes et établir les buts métiers de haut niveau (ou comprendre ceux déjà identifiés). Puisque les parties prenantes fournissent les exigences et les limites, il est très important de bien toutes les identifier. Les buts métiers aident à conserver la vision de ce qui doit être fait – les exigences recueillies durant l'Elicitation des Exigences aident à leur tour à atteindre ces buts métier. Aligner les exigences avec les buts métiers aide également à maîtriser le périmètre de la solution.

Les exigences initiales sont fournies par le client. Le client est une des parties prenantes clé du projet puisque ces besoins doivent être satisfaits.

Le client doit toujours être impliqué. Le but est de créer une compréhension mutuelle avec le client. Le fournisseur devrait toujours se mettre à la place du client.

Une des sources d'exigences la plus importante, est le contrat qui lie le fournisseur et le client. L'accord (contrat) doit formellement indiquer et décrire ce que le client veut. Il faut s'assurer que l'intérêt du client est central (ce qui implique que les fournisseurs ne devraient pas imposer une solution qu'ils préfèrent mais devraient analyser les besoins du client et recommander une solution qui réponde à ses besoins de la meilleure façon possible).

Le contrat doit être conforme aux ressources disponibles pour mettre en œuvre la solution et doit être basé sur : les estimations, les délais, les prix et les plans de projet. L'Ingénierie des Exigences fournit des informations d'entrée pour ces estimations.

Le contrat devrait inclure :

- Une brève description de la solution envisagée
- La liste des exigences par ordre de priorité
- Les critères d'acceptation pour chaque exigence
- La liste des produits à fournir (par exemple, la documentation, le code, le logiciel fonctionnant, les services, les processus)
- Le calendrier des échéances de développement et de livraisons du produit
- Les autres besoins et attentes tels que la technologie préférée, les besoins en ressources, etc.

Une autre source importante d'expression des principaux besoins du client se trouve dans la vision du projet. Pour chaque projet, la vision doit être à nouveau décidée. Une vision devrait définir les objectifs de haut niveau à atteindre. Il est capital d'avoir une définition claire de la vision du projet puisqu'elle fournit un moyen de vérifier si le projet a produit la valeur attendue ou pas.

Une vision se concrétise par des objectifs à atteindre dans le cadre d'un projet spécifique. Les buts doivent exprimer les objectifs métiers du projet et doivent être S.M.A.R.T. pour mesurer s'ils sont atteints à la fin du projet. S.M.A.R.T. est une méthode utilisée pour établir des objectifs et définir leurs critères de qualité. S.M.A.R.T. impose que tous les objectifs présentent les caractéristiques suivantes [G. T, Doran]:

- S – Spécifique
- M – Mesurable
- A – Atteignable
- R – Réaliste
- T – Temporellement défini

Lors de l'élicitation des exigences, il est important de s'assurer que les exigences satisferont à la fois à la vision projet et aux objectifs S.M.A.R.T. définis.

Les procédés suivants peuvent aider à établir la vision et les objectifs :

1. Trouver les sources des exigences
2. Analyser la situation (évaluation objective)
3. Evaluer la situation actuelle (facteurs subjectifs compris)
4. Déduire la vision / objectifs (sur la base des calculs coût bénéfice)

Afin de permettre d'identifier toutes les exigences et besoins, toutes les parties prenantes côté client comme côté fournisseur doivent être identifiées. Chaque partie prenante ou groupe de parties prenantes peuvent fournir de nouvelles exigences et influencer la conception de la solution envisagée. Si toutes les parties prenantes ne sont pas identifiées, il y a un risque que des exigences ou limitations importantes ne soient pas connues et pas prises en compte dans la conception. Des parties prenantes manquantes peuvent entraîner des demandes complexes de changement du produit lors des phases ultérieures du projet ou après la mise en production du système.

Les parties prenantes devraient être décrites dans les termes suivants : nom, fonction, disponibilité, domaine, objectifs et intérêts.

Certaines parties prenantes peuvent créer des groupes d'intérêts (par exemple les parties prenantes métier). Les groupes d'intérêts devraient être réunis car cela permet de gérer leurs exigences de manière plus efficace.

La procédure d'identification et d'évaluation des parties prenantes comprend les activités suivantes :

- Identifier les parties prenantes (par l'analyse des processus métier, l'identification des propriétaires de processus et de produits, l'analyse des structures organisationnelles et les marchés)
- Regrouper les parties prenantes en groupes et sélectionner des représentants des groupes spécifiques (si possible)
- Déterminer les relations entre parties prenantes
- Identifier les conflits potentiels
- Analyser les conflits et leurs sources et trouver des opportunités gagnant-gagnant
- Identifier les parties prenantes pouvant réduire les risques pour les impliquer davantage dans les activités de projet

- Identifier les points de vue des parties prenantes

Lorsque toutes les parties prenantes, les objectifs métiers et la vision sont connus, il est possible de commencer l’Elicitation des Exigences. Les techniques les plus courantes pour l’Elicitation des Exigences sont les suivantes :

- Questionnaires
- Entretiens
- Auto-enregistrements
- Sollicitation d’information des représentants des clients sur site
- Identification sur la base de documents existants
- Réutilisation (Réutiliser les spécifications d’un autre projet)
- Brainstorming
- Observations sur le terrain
- Apprentissage
- Ateliers
- Cas d’utilisation

Technique	Description	Application	Désavantages
Questionnaires	Un questionnaire peut comporter des questions ouvertes ou fermées. Une question ouverte exige de la personne qui répond qu’il formule sa propre réponse. Une question fermée exige un choix de la personne qui répond parmi un nombre possible d’options. Ces options devraient être mutuellement exclusives.	Pour confirmer ou détailler les exigences déjà connues ; organiser le contenu des exigences ; sélectionner des options pour les exigences / solution	Non applicable pour la collecte de connaissances implicites Faible taux de retour sans la motivation des personnes qui répondent Peut-être directif, ce qui empêche l’identification des besoins réels des utilisateurs

Technique	Description	Application	Désavantages
Entretiens	<p>Technique basée sur l'échange oral entre l'interviewer qui, au travers de questions posées à l'interviewé, obtient des informations sur des sujets spécifiques.</p> <p>Cette technique très interactive permet de modifier l'ordre des questions préalablement préparées en fonction des réponses de l'interviewé et de la situation.</p> <p>Les bons entretiens sont plus difficiles à mener que dans le cadre d'une conversation normale, il faut éviter de diriger les réponses de l'interviewé ou de les interpréter (par exemple : ne pas terminer les phrases de l'interviewé).</p> <p>L'interviewer posera des questions ouvertes pour obtenir des informations et des questions fermées pour confirmer ou invalider des options spécifiques.</p>	<p>Pour obtenir des informations sur des sujets spécifiques ; pour clarifier des exigences</p>	<p>Coût élevé en termes de temps passé</p> <p>Réponses difficilement reproductibles (difficile d'obtenir les mêmes réponses quand l'entretien est répété)</p>

Technique	Description	Application	Désavantages
Auto-enregistrements	<p>Avec cette technique, une partie prenante (par exemple un utilisateur) documente les activités qu’il exerce pour accomplir une tâche spécifique. En plus de documenter les activités “en l’état”, l’utilisateur décrit également les modifications, les désirs et les besoins.</p> <p>Cette technique est souvent associée avec des démonstrations ou des revues de documents.</p>	<p>Pour comprendre des procédures ou des processus complexes, spécialement quand l’ingénieur en charge des exigences n’est pas en mesure de voir l’utilisateur effectuer son travail.</p>	<p>Néglige les activités “automatisées”</p> <p>Dépend fortement de la motivation et de l’expérience de l’utilisateur</p>
Représentants des clients sur site	<p>Cette approche est l’une des méthodes d’Élicitation (et de validation) des Exigences la plus efficace, car elle permet au représentant de surveiller systématiquement les progrès, vérifier l’exactitude de la conception et fournir des commentaires et de l’information complémentaire si nécessaire.</p> <p>Avoir des représentants du client sur site est une des principales règles des méthodes agiles.</p>	<p>Collecter et gérer les exigences en approches Agiles ;</p> <p>Produire des exigences orientées utilisateurs qui peuvent être facilement acceptées.</p>	<p>Coûts élevés pour le client</p> <p>Coûts d’adaptation</p>

Technique	Description	Application	Désavantages
Cas d'utilisation	<p>Les cas d'utilisation permettent à l'ingénieur des exigences de recueillir les fonctionnalités de la solution du point de vue des Acteurs. Un Acteur peut être un utilisateur final ou un système.</p> <p>Les diagrammes de cas d'utilisation définissent les limites de la solution et montrent les connexions avec les systèmes et acteurs externes.</p>	Collecter et expliquer les exigences du point de vue de l'utilisateur ; clarifier et organiser la fonctionnalité attendue de la solution.	Pas applicable dans le cas de solutions qui consiste surtout en des exigences non-fonctionnelles.

Technique	Description	Application	Désavantages
<p style="writing-mode: vertical-rl; transform: rotate(180deg);">Identification des exigences sur la base de documents existants</p>	<p>Cette technique peut être utilisée quand il existe de la documentation qui peut aider à l’identification des exigences au sein de l’organisation. Des exemples de cette documentation peuvent inclure des modèles et cartes de processus, des descriptions de processus, des organigrammes, des spécifications de produits, des procédures (par exemple des procédures de travail), des normes et instructions, des modèles de documents, etc.</p> <p>Les exigences identifiées sont la base pour davantage d’Analyse des Exigences et doivent être détaillées et étendues à d’autres Exigences liées.</p>	<p>Pour collecter les exigences d’une solution qui couvrira les processus métier existants.</p>	<p>Ne s’applique pas quand il n’y a que des documents trop rudimentaires ou pas de document au sein de l’organisation.</p> <p>Ne s’applique pas lorsque la documentation n’est pas maintenue correctement (pas tenue à jour).</p>

Technique	Description	Application	Désavantages
Réutilisation (Réutiliser les spécifications d'un autre projet)	<p>Réutiliser la spécification d'un projet similaire peut-être fait quand une organisation a déjà terminé un ou plusieurs projets similaires au projet en cours. Une spécification des exigences préparée pour un ou des projet(s) précédent(s) peut être utilisée dans un autre projet afin de raccourcir la durée d'Analyse des Exigences et l'effort de préparation de la documentation, ce qui devrait permettre de commencer l'exécution du projet plus tôt.</p> <p>Dans la plupart des cas, seules certaines parties des spécifications existantes peuvent être utilisées pour un nouveau projet. Pour être réutilisée, la documentation doit être conforme aux besoins et exigences actuels et devrait donc être vérifiée et ajustée en conséquence.</p>	<p>Pour les projets visant à fournir des produits sur mesure pour des clients spécifiques ; pour raccourcir l'Ingénierie des Exigences dans des projets développant des solutions similaires.</p>	<p>Coûts élevés pour le premier projet.</p> <p>Exigences trop détaillées, résultant en une vaste et coûteuse gestion du changement.</p>

Technique	Description	Application	Désavantages
Observations sur le terrain	<p>L’observation terrain consiste à observer les activités et processus des utilisateurs et identifier les exigences du système sur cette base. L’observation sur site est menée en regardant les utilisateurs travailler puis documenter les processus, tâches et résultats.</p>	<p>Pour recueillir des exigences précises en observant les utilisateurs au travail ; pour éviter les problèmes que les parties prenantes ont pour améliorer les processus métier ou systèmes existants.</p>	<p>Les cas exceptionnels peuvent être oubliés</p> <p>Pas applicable dans certaines conditions (par exemple, pour raisons de sécurité ou juridique)</p>
Apprentissage	<p>Le but de l’apprentissage est de collecter des exigences d’un client, surtout lorsque les processus et activités réalisés par le personnel du client ne sont pas faciles à décrire en utilisant d’autres techniques, telles que les entretiens, ou lorsque le client a des difficultés à exprimer les exigences.</p> <p>L’apprentissage est un processus de formation au métier du client. Le client, qui connaît le mieux la façon de faire un travail spécifique, enseigne à l’ingénieur des exigences, comme un maître à son élève.</p>	<p>Pour comprendre les processus métier complexes pour être en mesure de proposer la meilleure solution ; pour aider à surmonter la difficulté que les employés du client peuvent avoir à penser de façon abstraite et exprimer leurs tâches en mots.</p>	<p>Le coût élevé</p> <p>Temps nécessaire important</p> <p>Pas applicable dans les environnements dangereux</p>

Technique	Description	Application	Désavantages
Ateliers	<p>Un atelier est une sorte de réunion qui se concentre sur un sujet spécifique (défini et annoncé aux participants au préalable), qui implique habituellement les parties prenantes représentant les différentes régions et/ou domaines pour une courte et intense durée.</p> <p>Les ateliers impliquent des personnes qui ont des points de vue différents sur un problème donné et aident à déterminer et décrire des exigences provenant de différentes perspectives.</p> <p>Les ateliers sont une des pratiques clés des méthodes agiles puisqu'ils impliquent toutes les parties prenantes dans le développement de la Product Backlog. Des ateliers bien préparés et bien menés, permettent de mieux gérer des situations comme :</p> <ul style="list-style-type: none"> - améliorer les relations d'équipes éloignées géographiquement, - permettre de travailler en groupes restreints dans les équipes où le consensus est difficile à obtenir. 	<p>Pour identifier des exigences afin d'établir le périmètre de la solution ; pour découvrir les exigences cachées (c.-à-d., les exigences qui ne sont pas directement formulées ou dont les parties prenantes n'ont même pas conscience mais qui sont nécessaires pour accomplir certains de leurs besoins ou exigences de plus haut niveau) ; pour développer ou détailler des exigences d'un domaine récemment identifié ; pour définir les priorités des exigences ou atteindre un consensus lorsqu'il s'agit de s'accorder sur les exigences (approbation) ; pour identifier et résoudre des conflits potentiels entre exigences des parties prenantes ; pour revoir les résultats des processus ou activités spécifiées et résoudre les problèmes qui peuvent être apparus.</p>	<p>Difficile dans le cas d'équipes éloignées géographiquement</p> <p>Disponibilité de toutes les personnes requises pour l'atelier</p> <p>Le consensus n'est pas forcément facile à atteindre en atelier, et la discussion peut bloquer sur des problématiques, rendant le procédé long et démotivant pour les participants.</p>

Technique	Description	Application	Désavantages
Brainstorming	Le brainstorming est une technique communément utilisée pour obtenir les exigences liées à des domaines de l'activité de l'organisation ou des fonctionnalités prévues du système qui sont nouvelles ou mal connues. Cela permet de recueillir beaucoup d'idées de différentes parties prenantes en peu de temps et à moindre coût. Pendant une session de brainstorming, les participants soumettent des idées ou des concepts concernant un problème donné.	Pour résoudre des conflits dans les exigences ; pour définir les différentes options d'une solution ; pour résoudre les problèmes métier (par exemple, faible efficacité d'un processus).	Difficile avec des participants peu motivés Difficile à appliquer dans les équipes distribuées

Table 2 - Techniques d'Elicitation des Exigences

[Pour les organismes de formation : fournir des exemples d'applications d'au moins trois techniques différentes d'Elicitation des Exigences.]

Pour obtenir de vrais résultats et éviter des lacunes dans les exigences, une combinaison des techniques décrites ci-dessus est utilisée.

Comme indiqué précédemment, les exigences fonctionnelles précisent les fonctions du système telles qu'elles sont perçues par l'utilisateur final. Les exigences fonctionnelles décrivent également les déclencheurs du processus tels que les actions des utilisateurs ou les entrées / sorties de données qui font démarrer le processus métier.

Lors de l'élicitation des exigences, il est important non seulement de poser des questions à propos des fonctions mais également des attributs qualité. Les exigences non fonctionnelles (ENF) décrivent les attributs qualité de l'ensemble du système ou de ses composants et fonctions spécifiques. Ils peuvent limiter la solution, par exemple, en exigeant des critères d'efficacité spécifiques. Les exigences non fonctionnelles sont difficiles à décrire et sont souvent exprimées de façon vague, voire pas du tout documentées, ce qui les rend difficile à tester. Une attention particulière devrait

être portée aux exigences non fonctionnelles à toutes les étapes du processus EE pour être sûr qu'ils sont exprimés clairement et sont mesurables.

Les exigences non fonctionnelles peuvent décrire les différents aspects de la performance de la solution. Selon la norme ISO/IEC 25000 (anciennement ISO 9126), les types d'exigences non fonctionnelles définis sont les suivants :

- Fiabilité
- Utilisabilité
- Efficacité
- Maintenabilité
- Portabilité

Les exigences non fonctionnelles sont particulièrement importantes car elles spécifient des critères qui peuvent être utilisés pour évaluer le bon fonctionnement d'un système ; elles ont par conséquent un grand impact sur la satisfaction du client utilisant le produit.

Les exigences fonctionnelles doivent fournir des fonctions ; les exigences non fonctionnelles déterminent comment ces fonctions peuvent être utilisées facilement et efficacement.

[Pour les organismes de formation : fournir des exemples d'exigences fonctionnelles et non-fonctionnelles selon la norme ISO / IEC 25000.]

Les exigences, une fois élicitées, doivent être correctement documentées afin de permettre leur traçabilité et leur analyse.

Les exigences doivent être clairement et précisément spécifiées. Elles devraient être mesurables afin de s'assurer qu'elles sont vérifiables et que leur mise en œuvre puisse être correctement contrôlée. Il est important de se rappeler que le langage a des limites et des inconvénients. Cela peut entraîner des descriptions d'exigences peu claires ou ambiguës. Par conséquent, des normes et modèles appropriés devraient être utilisés autant que possible. Les normes permettent une compréhension partagée et les meilleures pratiques pour écrire des spécifications alors que les modèles limitent le langage qui peut être utilisé.

En plus des normes et des modèles, le vocabulaire est un des outils importants pour faciliter la communication entre les différentes parties prenantes et introduire un certain degré de contrôle sur l'ambiguïté du langage naturel.

La description des exigences doit répondre à des critères qualité (voir 1.1.4 Qualité des besoins).

En fonction du niveau d'abstraction, les exigences peuvent être décrites de façon plus ou moins détaillée. Dans certains modèles de développement, les exigences métier peuvent être écrites sous la forme de cas d'utilisation de haut niveau (par exemple, Rational Unified Process), ou des user stories (approches agiles).

En général, la structure typique d'une expression d'exigences devrait couvrir les aspects suivants :

- L'utilisateur – qui souhaiterait cette exigence ?
- Le résultat – quel résultats les parties prenantes cherchent elles ?
- L'objet – quel objet l'exigence aborde-t-elle ?
- Le qualificatif - quel est le qualificatif mesurable ?

Par exemple, la déclaration des exigences suivantes : « L'agent d'assurances doit avoir des informations sur tout nouveau produit un jour avant le lancement du produit. » couvre les quatre éléments énumérés ci-dessus de la manière suivante :

- L'agent d'assurance ← qui
- Doit avoir de l'information sur ← quel résultat
- Tout nouveau produit ← quel objet
- Un jour avant le lancement du produit ← qualificatif

Des exigences plus précises (exigences solution/système) peuvent être décrites sous forme de cas d'utilisation du système, souvent avec des scénarios ou sous forme de « user stories » dans les approches agiles. En général, les exigences de solution/systèmes couvrent les aspects suivants :

- Processus concerné – à quel processus métier l'exigence fait-elle référence ? Cet élément se concentre habituellement sur les fonctionnalités et détermine les entrées et sorties
- Activité de la solution – comment l'activité est-elle déclenchée ? Est-ce une activité indépendante du système, ou une interaction avec l'utilisateur, ou une interface avec d'autres systèmes/processus ?
- Obligation juridique – quel est l'obligation juridique de l'exigence ? Des mots-clés tels que devrait, il faut etc. sont souvent utilisés pour le décrire.
- Contraintes logiques et de temps – y-a-t-il des conditions de limites applicables à l'exigence ?

[Pour les organismes de formation : discuter sur des exemples de descriptions d'exigences métier et d'exigences de solutions/systèmes.]

Les exigences identifiées et analysées sont généralement documentées sous la forme d'une Spécification d'Exigences (voir 5.4 Spécification des Exigences).

5.3 Analyse des Exigences

80 minutes

5.3.1 Introduction

L'étape qui suit l'Elicitation des Exigences est l'Analyse des Exigences. Le principal objectif de l'analyse est de créer une solution métier qui mettra en œuvre les exigences. Créer la solution métier revient à détailler les besoins clients en exigences de systèmes/solution jusqu'aux exigences de produit-composant. Cela comprend également la détection et la résolution de conflits entre les exigences de mêmes niveaux et de niveaux différents, et établir les limites de la solution y compris la façon dont elle va interagir avec son environnement.

La procédure simplifiée d'Analyse itérative des Exigences contient les étapes suivantes :

1. Analyse des besoins
2. Description de la solution
3. Estimation de la charge et priorisation
4. Acceptation des exigences

La première étape vise à analyser les besoins des parties prenantes pour être en mesure de proposer une solution répondant aux besoins. L'Analyse des Exigences comprend souvent des activités de vérification et de validation pour s'assurer que les exigences sont bien comprises et approuvées par les parties prenantes. Une partie importante de l'Analyse des Exigences est de vérifier la clarté et la cohérence des exigences et de résoudre les conflits entre elles.

La description de la solution est habituellement effectuée à partir de modèles. En général, il existe trois principaux types de modèles d'analyse : les modèles d'exigences, les modèles de solution, et les modèles conceptuels. Les modèles sont développés grâce à des méthodes d'analyse appropriées.

L'étape suivante dans l'Analyse des Exigences porte sur l'estimation des charges et la priorisation des exigences.

La dernière étape de l'Analyse des Exigences a pour objectif l'acceptation des exigences.

5.3.2 Estimation des coûts (charge)

Les estimations de charge lient l'Ingénierie des Exigences avec la Gestion de Projet. Les premières estimations de charge sont basées sur les exigences de haut niveau et sont utilisées pour estimer le coût du projet. Pendant le développement de la solution, lorsque les exigences évoluent, les estimations de coûts initiales sont utilisées pour contrôler le projet et – dans le cas de changements – pour estimer la charge et le coût des changements.

Les estimations sont généralement axées sur les points suivants :

- Coûts
- Temps
- Exigences

Le coût du projet dépend de divers facteurs, incluant :

- Le type de projet
- La maturité des processus
- Les approches de conception et de test, les méthodes et les outils
- La technologie utilisée
- La complexité de la solution envisagée
- Les objectifs de qualité (par exemple, le niveau souhaité de qualité du produit)
- Les qualifications de l'équipe
- La répartition de l'équipe
- L'expérience

La précision de l'estimation de charge dépend de l'état d'avancement du projet et de sa maturité.

En général, l'estimation de charge peut être réalisée en utilisant des raisonnements par analogie et des procédures algorithmiques. Ces deux approches exigent que les procédures d'estimation soient basées sur des données historiques et des conditions de cadrage.

Les raisonnements par analogie sont basés sur des comparaisons avec d'autres projets. L'estimation est basée sur l'expérience et pas sur des formules mathématiques. Avec cette technique, le projet à estimer est comparé avec des projets antérieurs. La comparaison peut comprendre :

- Le nombre et la complexité des exigences

- La stabilité métier
- Le périmètre de la solution
- La technologie utilisée
- Les particularités du personnel (par exemple les compétences, l'expérience)

Quelques exemples de procédures de type analogique :

- "Planning Poker"
- Méthode Delphi

Un exemple de procédure analogique peut être trouvé dans les projets Agile. Dans ces projets, surtout lorsqu'ils sont gérés avec Scrum, il existe une méthode appelée "Planning Poker". Cette méthode est utilisée pour obtenir un consensus parmi l'équipe. La capacité de l'équipe est alors mesurée par ce qu'on appelle la « Burn Down Rate ». Des sessions de rétrospective sont menées après chaque sprint et les points estimés sont comparés à la durée réelle. Cela améliore la capacité de l'équipe à estimer.

La méthode Delphi est un autre exemple de technique analogique. Il s'agit d'une technique de communication structurée utilisée pour effectuer des prévisions interactives. Elle réunit un comité d'expert. Les experts sont invités à répondre à des questionnaires en un certain nombre de tours. Après chaque tour, les prévisions faites par les experts sont résumées ainsi que les raisons de leurs décisions. Les experts révisent ensuite leurs estimations antérieures en tenant compte des réponses des autres membres de leur comité.

On pense que, durant ce processus, l'étendue des réponses va diminuer et que le groupe va converger vers la réponse « correcte ». Le processus est arrêté à l'atteinte d'un critère d'arrêt prédéfini. Les scores moyens des derniers tours déterminent les résultats.

La deuxième approche d'estimation de charge est une procédure algorithmique. Dans ce cas, les charges sont calculées sur la base de paramètres. Les paramètres peuvent décrire le produit (volume, durée), les conditions aux limites (efficacité), etc. Des exemples de procédures algorithmiques incluent :

- Analyse des Points de Fonctions
- Analyse des Points de Cas d'Utilisation

L'analyse des Points de Fonctions nécessite de compter les dits points de fonctions de la solution envisagée. Un point de fonction est une unité de mesure qui exprime la quantité de fonctionnalités métier fournies par un système d'information à un utilisateur. Le coût d'un seul Point de Fonction est calculé sur la base de projets antérieurs. Les fonctions identifiées sont pondérées en fonc-

tion de trois niveaux de complexité ; le nombre de points assigné à chaque niveau varie en fonction du type de Point de Fonction.

L'Analyse des Points de Cas d'Utilisation est assez similaire à l'Analyse des Points de Fonction. Cette méthode nécessite d'analyser d'autres éléments comme le nombre et la complexité des cas d'utilisation ainsi que les acteurs de la solution.

5.3.3 Priorisation

Une autre activité importante de l'Analyse des Exigences est la priorisation. Prioriser établit l'importance relative de mise en œuvre d'une exigence. Cela permet à l'équipe de terminer les exigences les plus importantes en premier. La priorisation permet également le développement incrémental, puisque les exigences peuvent être regroupées et livrées par priorité dans les itérations.

La procédure pour établir les priorités des exigences comprend les activités suivantes :

1. Regroupement des exigences – qui comprend l'identification des exigences qui influencent ou sont dépendantes les unes des autres (par exemple, les exigences qui créent un ensemble complexe de fonctions).
2. Analyse des exigences – analyser les exigences regroupées par les parties prenantes concernées afin de s'entendre sur le niveau d'importance et établir les priorités des exigences.
3. Création du plan de gestion des exigences – établir un plan où les exigences de haute priorité sont à développer les premières et attribuer les responsabilités pour les mettre en œuvre.
4. Planification du système de tests des incréments – concevoir des cas de test pour tester chaque incrément de solution sur la base des priorités des exigences pour cet incrément.

Une approche commune pour prioriser est de regrouper les exigences en catégories de priorité. Souvent, une échelle à trois niveaux est utilisée (c.-à-d. Haut, Moyen, Bas). Comme ces échelles sont subjectives et imprécises, toutes les parties prenantes doivent s'accorder sur le sens de chaque niveau de l'échelle utilisée. La définition de la priorité doit être clairement indiquée et doit être un attribut clé de chaque exigence.

Pour les organismes de formation : expliquer la procédure de priorisation avec des exemples. Expliquer le sens de la priorisation des exigences pour la planification de version / itérations.

5.3.4 Accord sur les exigences

L'acceptation des exigences (également appelé accord ou approbation des exigences) est le prochain élément de l'Analyse des Exigences. Il s'agit d'un accord formalisant que le contenu et le périmètre des exigences sont exacts et complets. Il est très important de s'assurer que les exigences ont été acceptées lors des différentes phases de développement de la solution puisqu'un accord formel est une base pour la poursuite des activités. L'accord sur les exigences de haut niveau (exigences métier) devrait être obtenu avant le début du projet. L'accord sur les exigences détaillées (exigences de solution/système et exigences de composant/fonction) devrait être obtenu, par signature, avant de passer à la phase de mise en œuvre. Obtenir l'approbation des exigences est généralement la tâche finale de l'Analyse des Exigences et de la phase de spécification du projet.

L'approbation des exigences implique généralement les principales parties prenantes du projet y compris :

- Les Chefs de Projets
- Les représentants métier des clients
- Les Analystes Métier et Systèmes
- Les Ingénieurs en exigences
- Les représentants de l'Assurance Qualité, les équipes de tests et de développement

L'un des objectifs de l'acceptation des exigences est de s'assurer que les exigences sont stables et que tout changement est géré par des Demandes de Changement. Un accord formel réduit le risque d'introduire de nouvelles exigences pendant ou après la mise en œuvre.

L'acceptation formelle peut également réduire le risque de malentendu entre le client et le fournisseur en ce qui concerne le périmètre du projet puisque toutes les exigences ou les spécifications d'exigences/solutions doivent être examinées avant la signature.

Les accords sur les exigences sont considérés comme complets quand toutes les parties prenantes pertinentes du projet ont signé la documentation des exigences.

L'achèvement de l'acceptation des exigences devrait être communiqué à l'équipe projet et est généralement un jalon important du projet.

5.3.5 Modélisation de la solution

La Modélisation de la Solution fait souvent partie de l'Analyse des Exigences et des Spécifications puisque son but est de développer des modèles des problèmes réels qu'il faut résoudre par des solutions métier spécifiques. La Modélisation de la Solution peut également être effectuée pendant l'Elicitation des Exigences car certaines des techniques d'élicitation utilisent différentes notations ou approches de modélisation (par exemple, l'Elicitation des Exigences par des cas d'utilisation UML). Le principal défi pour la Modélisation de la Solution est de choisir et développer des diagrammes appropriés qui peuvent former un modèle décrivant la solution métier sous différents points de vue. Ces modèles ne devraient pas simplement représenter la solution métier, mais doivent aussi être compréhensibles par le client et les représentants des fournisseurs.

La Modélisation de la Solution peut utiliser plusieurs types de modèles, mais en général il existe 3 niveaux fondamentaux :

- Modèle d'exigences – décrit le domaine du problème et est généralement conçu dès les premières étapes du projet. Ce modèle sert essentiellement pour l'Analyse des Exigences et l'estimation de charge et fournit une base au modèle de la solution.
- Modèle de solution – décrit le domaine de la solution à partir de différents points de vue et détermine la forme d'implémentation des exigences fonctionnelles et non-fonctionnelles. Le modèle de solution métier fournit une base au modèle de conception de solution.
- Modèle conceptuel – également appelé modèle de domaine. Un modèle conceptuel représente les concepts (entités) et les relations entre eux. L'objectif de la modélisation conceptuelle est de clarifier et exprimer le sens des termes et concepts utilisés par les experts du domaine pour résoudre le problème métier, et établir les relations exactes entre les différents concepts.

Différents autres modèles peuvent être utilisés dans les niveaux ci-dessus en fonction du point de vue spécifique à présenter. Les perspectives communes applicables à la modélisation du problème ou du domaine de la solution sont les suivantes :

- Vue utilisateur (par exemple, modélisée à partir des cas d'utilisation)
- Vue logique (par exemple, modélisée à partir d'exigences fonctionnelles)
- Vue processus (par exemple, modélisée à partir des modèles de communication, des modèles d'interactions, ou des exigences non fonctionnelles spécifiant l'efficacité des processus métier)
- Vue d'implémentation (par exemple, généralement modélisée à partir des composants de la solution)

- Vue installation (par exemple, modélisée à partir des modèles d'intégration et de l'architecture de la solution)

Différents niveaux de modélisation et différentes vues de la solution peuvent être décrits par différents diagrammes. Pour avoir l'image complète de la solution, une combinaison de différents points de vue est habituellement utilisée. Il en résulte une utilisation de différents diagrammes décrivant le modèle de la solution à partir de différentes perspectives. Quelques exemples de modèles d'analyse sont présentés dans le tableau ci-dessous (Tableau 4 exemples de méthodes et modèles d'analyse).

Méthode d'analyse	Modèles d'analyse
Analyse de contexte	Diagramme de contexte Diagramme de déploiement (UML)
Analyse logique	Décomposition fonctionnelle Diagramme des exigences (SysML)
Analyse de décision	Diagramme état transition (UML)
Analyse de données	Diagramme entité - relation (ERM) Diagramme de classe (UML)
Analyse de cas d'utilisation	Diagramme de cas d'utilisation (UML)
Analyse de flux de processus	Diagramme de processus métier (BPMN) Diagramme d'activité (UML) Diagramme de communication (UML)

Table 3 - Exemple de méthodes d'analyses et diagrammes

Pour les organismes de formation : expliquer les bénéfices et application des modèles de base d'analyse (modèles de contexte, décomposition fonctionnelle, modèles de processus métier) et fournir des exemples d'utilisation de ces diagrammes.

Une des méthodes les plus communes de modélisation de solutions logicielles est UML (Unified Modeling Language). UML fournit plusieurs types de diagrammes pour décrire les différentes vues de la solution. UML est une notation unifiée pour l'analyse et la conception de systèmes. UML fournit différents diagrammes divisés en diagrammes de comportement et de structure, où les diagrammes de comportement décrivent les caractéristiques comportementales d'un système ou

d'un processus métier, et les diagrammes de structure décrivent les éléments structurels composant un système ou une fonction.

Type de diagramme UML	Application
Diagrammes de comportement	
Diagrammes d'activité	Modéliser les comportements d'un système, et la façon dont ces comportements sont reliés dans un flux global du système. Habituellement utilisé pour modéliser des procédures, des flux de processus et des « workflow »
Diagrammes de cas d'utilisation	Capturer les cas d'utilisation et les relations entre les acteurs et le système. Décrire les exigences fonctionnelles du système, la manière dont les opérateurs externes interagissent aux frontières du système, et la réponse du système.
Diagrammes de machine à états	Montrer comment un élément peut évoluer entre états, classer son comportement selon des lois de déclenchements et des contraintes de garde.
Chronogrammes	Définir le comportement de différents objets selon une échelle de temps. Fournir une représentation visuelle d'objets changeant d'état et interagissant dans le temps
Diagrammes de séquence	Représentation structurée de comportement en séries d'étapes séquentielles dans le temps Utilisé pour décrire le « workflow », le passage de messages et comment, en général, les éléments coopèrent pour atteindre un résultat.
Diagrammes de comportement	
Diagrammes de communication	Montrer les interactions entre les éléments durant le temps d'exécution, en visualisant les relations entre les objets
Diagrammes d'ensemble d'interactions	Visualiser la coopération entre d'autres diagrammes d'interactions pour illustrer un flux de contrôle servant un but

Type de diagramme UML	Application
Diagrammes de structure	
Diagrammes de classe	Décrire la structure logique du système, les classes et les objets créant le modèle, décrivant ce qui existe et quels attributs et comportements il possède. Peut être utilisé pour modéliser des données.
Diagrammes d'objets	Présenter des instanciations de classes et leurs relations à un moment donné.
Diagrammes de package	Décrire l'organisation des éléments du modèle en paquet et les dépendances entre eux. Peut être utilisé pour organiser les exigences selon leurs types/niveaux d'abstraction.
Diagrammes de composants	Présenter les parties de logiciel créant un système ainsi que leur organisation et dépendances.
Diagrammes de déploiement	Décrire l'architecture d'exécution du système.
Diagrammes de structure composite	Refléter la collaboration interne des classes, interfaces et composants (et leurs propriétés) pour décrire une partie d'une fonctionnalité.
Diagrammes de profils	Permettre d'adapter un méta-modèle UML pour différentes plateformes (telles que J2EE ou .NET) ou domaines (comme le temps-réel ou la modélisation de processus métier).

Table 4 - Les diagrammes UML et leurs applications

Pour modéliser des solutions plus complexes, en particulier dans le domaine de l'ingénierie des systèmes, une autre notation de langage unifié peut être utilisée – SysML (System Modeling Language). SysML permet de modéliser une large gamme de systèmes qui incluent matériel, logiciel, information, processus, personnel et infrastructures. SysML réutilise sept des diagrammes UML et fournit deux nouveaux diagrammes : un diagramme d'exigences qui capture les exigences fonctionnelles, de performance et d'interface, et un diagramme paramétrique qui définit les contraintes de performance et quantitatives.

Pour modéliser les flux de processus métier et la communication entre les acteurs métier, d'autres notations de modélisation peuvent être utilisées – par exemple BPMN (Business Process Modeling Notation) qui est habituellement utilisée dans les phases amont de l'analyse des exigences ou durant l'analyse des processus métier au sein de l'organisation du client.

Pour modéliser les aspects de l'interface utilisateur de la solution (particulièrement dans le cas des solutions logicielles), la modélisation d'IHM peut être utilisée. Habituellement effectuée par l'intermédiaire de prototypage, elle fournit un moyen de concevoir et de tester des éléments de l'interface utilisateur. La modélisation d'IHM est particulièrement utile dans les cas où les exigences ne sont pas claires ou peuvent être implémentées de plusieurs façons différentes. Elle permet aux acteurs de choisir la meilleure option d'implémentation.

Les différents types de prototypage peuvent être les suivants [IBUQ] :

- Prototypes verticaux : réduction à peu de fonctions individuelles mais détaillées.
- Prototypes horizontaux : autant de fonctions intégrées que possible, mais non-fonctionnelles (principalement utilisé pour le test d'interface utilisateur).
- Prototypes de scénario : toutes les fonctions sont simulées pour une tâche spécifique en utilisant une combinaison de prototypes verticaux et horizontaux.

En fonction de l'objectif à atteindre, des prototypes sont créés selon différentes formes et variantes [IBUQ] :

- Prototypes de faible-fidélité (lo-fi) – faible similarité avec le produit final ; examiner l'intérêt de l'idée
- Prototypes d'haute-fidélité (hi-fi) – haute similarité ; revue des détails et
- Formes hybrides – les prototypes de faible-haute-fidélité (lo-hi-fi), par exemple, des simulations interactives utilisant le HTML ou une présentation PowerPoint

[Pour les organismes de formation : expliquer les bénéfices et l'application de diagrammes UML de base (diagrammes d'activité, diagrammes de cas d'utilisation, diagrammes de machine d'état et diagrammes de classe) et fournir des exemples d'utilisation de ces diagrammes.]

5.4 Spécification des Exigences

60 minutes

Le développement des exigences n'est pas seulement un processus de découverte et d'analyse des exigences, il est aussi un processus de facilitation d'une communication efficace de ces exigences entre les différentes parties prenantes. La manière dont les exigences sont documentées joue un rôle important pour assurer qu'elles puissent être lues, analysées, changées et validées. Documenter les exigences est appelé : spécification des exigences.

Une spécification est définie comme un ensemble explicite d'exigences devant être satisfaites par un matériel, produit ou service ; par conséquent, elle peut être considérée comme une sorte de contrat décrivant le comportement et les caractéristiques de la solution. La spécification sert à tracer et à gérer les exigences. Dans la spécification, les exigences sont spécifiées d'une manière structurée et elles sont modélisées séparément (les exigences sont modélisées « indépendamment les unes des autres », au sens où les exigences de haut-niveau sont ventilées vers les niveaux inférieurs dans lesquels chaque exigence individuelle constitue une entité indépendante pouvant être par la suite développée et tracée). La spécification est un document sur lequel est basé un agrément formel sur les exigences devant être implémentées dans le système planifié (ou dans d'autres formes de solution), par conséquent la spécification des exigences peut être considérée comme une partie ou le résultat final de l'analyse des exigences.

Le terme « spécification » peut être utilisé dans le contexte des exigences de besoin et des exigences de solution.

La spécification des exigences de besoin décrit le périmètre du problème (une proposition de solution métier pour un problème métier donné, besoin ou objectif, une nouvelle fonctionnalité, etc..) et elle contient au moins les informations suivantes :

- Des exigences métier avec leur critère d'acceptation
- Des limites et des hypothèses

La création de la spécification des exigences du client devrait être la tâche du client. Toutefois, dans certains cas, le fournisseur peut aider à préparer la spécification des exigences de besoin.

Une spécification d'exigences n'est pas censée être un document formel de « spécification ». Par exemple, il peut être un « backlog » de « sprint » ou un jeu d'exigences maintenu dans un outil de gestion d'exigences.

La création d'autres types de spécification d'exigences telle que des exigences système, des exigences logicielles, des exigences de sûreté de fonctionnement, des exigences de sécurité, des exigences environnementales, des exigences légales, etc. sont des tâches pour d'autres rôles.

La spécification de solution décrit la solution de différents points de vue. Les types spécifiques de spécification de solution incluent : les spécifications des exigences fonctionnelles, les spécifications d'exigences système ou les spécifications d'exigences logicielles.

Par exemple, une spécification fonctionnelle est un document qui décrit clairement ce que doit faire la solution. La spécification fonctionnelle est la base pour un développement ultérieur de produit et pour le test. Elle doit fournir des informations précises concernant tous les aspects fonctionnels du logiciel devant être implémentés. Une spécification fonctionnelle ne décrit pas comment implémenter les fonctions souhaitées du produit logiciel et elle ne prescrit pas quelle technologie doit être utilisée. Par contre, elle focalise sur les fonctionnalités et les interactions entre l'utilisateur et le produit.

La spécification des exigences de besoin et la spécification des exigences de solution peuvent être écrites toutes-deux sous la forme de cas d'utilisation (cas d'utilisation métier ou cas d'utilisation système, selon le but et l'audience cible de la spécification).

Un autre type de spécification est une « User Story ». Les « User Stories » sont souvent utilisées dans les méthodologies de développement « Agile ». Les « User Stories » sont un moyen rapide de prendre en charge les exigences client/utilisateur. L'intention de la « User Story » est d'être capable de répondre plus rapidement et avec moins de frais à des exigences changeant rapidement dans la réalité.

Une « User Story » décrit la fonctionnalité qui sera utile pour le client. Elle est composée de 3 aspects [Cohn] :

- Une description écrite de l'histoire utilisée pour la planification et comme un memento (habituellement sous la forme d'une déclaration « En tant que [rôle utilisateur final], je veux [le désir] pour [le justificatif] »)
- Des conversations sur l'histoire qui servent à en étoffer les détails
- Des tests qui véhiculent et documentent les détails et sont utilisés pour déterminer à quel moment l'histoire est complète

Les « User Stories » sont souvent utilisées avec des archétypes de personnage représentant un type spécifique de rôle d'utilisateur final.

Le travail de spécification peut être soutenu par des standards qui fournissent des directives sur la façon de rédiger un document spécifique, son contenu type, ou les meilleures pratiques. Les standards qui peuvent être communément utilisés pour la création de spécification sont les suivants :

- IEEE 830 : Pratiques recommandées pour la spécification d'exigences logicielles, également connu comme standard pour SRS (Software Requirements Specification).

- IEEE 1233 : Guide pour le développement des spécifications des exigences système (également connu comme standard pour SyRS)
- IEEE 1362 : Guide pour les technologies de l'information – Définition de système (également connu comme Concept d'opérations, ConOps).

En général, une spécification sert d'activité de formalisation des résultats de l'analyse des exigences.

La procédure pour créer une spécification de solution inclut habituellement les activités suivantes :

Définition de la vision et objectif

- Quel est l'objectif de la solution ou le périmètre spécifique de la solution qui est documenté dans la spécification de la solution ?

Identification des parties prenantes

- Qui utilisera la solution ?
- Qui est responsable des périmètres spécifiques de la solution et peut supporter ou impacter la création de la spécification de la solution ?

Détermination des exigences

- Quelle sont les exigences métier (business) de haut-niveau qui devraient être couvertes dans la spécification ?
- Quelles sont les priorités des exigences ?

Spécification d'exigences structurée

- Quelles sont les dépendances entre les exigences ?
- Comment les exigences peuvent-elles être organisées ?

Description de l'environnement du système

- Quel est le contexte de la solution ?

Détermination de la solution

- La définition de la solution et de sa portée avec les aspects pertinents extérieurs, mais influençant la portée, telle que les interfaces avec les systèmes extérieurs

Analyse des exigences

- Analyse détaillée des exigences (décomposition des exigences métier en solution/système de bas-niveau ou exigences de fonction/composant)
- Déterminer et résoudre les conflits potentiels entre les exigences
- Mapper les exigences métiers vers les exigences de solution/système et créer une structure pour la traçabilité

Modélisation du problème métier

- Modéliser le problème métier devant être résolu par la solution (incluant la modélisation ou l'amélioration des processus métier existants, si applicable). Des notations de modélisation de processus métier sont utilisées communément pour décrire des modèles métier (par exemple, BPMN)
- Comment la solution sera construite dans l'infrastructure métier existante ?

Modélisation de la solution

- Modéliser la solution (qui devrait inclure les caractéristiques du logiciel, du matériel, des services, des processus, etc. dépendant de la spécification du type de solution)
- Utiliser des notations de modélisation pour présenter la solution de différents points de vue (par exemple UML)

La procédure, ci-dessus, implique un certain nombre de parties prenantes qui supportent le travail de spécification dans différents périmètres et formalisent les résultats du processus d'analyse des exigences. Des étapes individuelles réalisées dans la procédure peuvent être faites en parallèle, itérativement ou même ignorées selon l'information déjà collectée et l'approche d'Ingénierie des Exigences sélectionnée (par exemple, dans les méthodologies Agile, la procédure pour la spécification de solution n'est pas aussi formelle puisque les exigences et solutions sont habituellement décrites de manière succincte). La sortie de la procédure, la spécification de solution, sert de point de départ pour la conception logicielle, matérielle et de base de données. Elle décrit la fonction

(spécifications fonctionnelles) du système, la performance (spécifications non-fonctionnelles) du système et les contraintes opérationnelles et d'interface utilisateur.

La documentation du produit, incluant les spécifications d'exigences ou de solutions, devrait être créée suivant différents degrés de formalisation :

- Non-formelle
- Semi-formelle
- Formelle

Une approche non-formelle pour écrire une spécification signifie que le document est écrit en langage naturel, sans aucune notation formelle. Cette approche peut être utilisée quand les lecteurs n'ont aucune expérience avec les notations formelles ou les langages techniques de spécification et auraient des difficultés dans la compréhension du contenu du document. La principale faiblesse de cette approche est qu'elle est ambiguë et qu'elle peut conduire à mauvaise compréhension et surinterprétation. De plus, la spécification non-formelle n'est pas une bonne base pour l'implémentation et le test puisqu'elle n'est pas suffisamment claire et précise.

Une spécification semi-formelle inclura des notations formelles et elle sera bien structurée. Habituellement, de telles spécifications sont basées sur des modèles de document spécifiques (souvent dérivés des standards pertinents). Les spécifications semi-formelles peuvent exprimer des exigences sous la forme de modèles et elles utilisent un langage naturel formalisé. Une des notations les plus communément utilisées pour la documentation semi-formelle est UML.

Une spécification formelle est une description mathématique et/ou algorithmique d'une solution qui peut être utilisée pour développer une implémentation. Une spécification formelle décrit ce que le produit devrait faire et habituellement elle ne décrit pas comment il devrait être fait. Comme elle est basée sur des formules mathématiques ou un langage spécifique (par. ex. : Langage de spécification VDM (VDM-SL)), et qu'elle est plus difficile à appréhender, la spécification formelle est rarement utilisée, de plus elle requière des connaissances spécifiques.

5.5 Vérification et Validation des Exigences

40 minutes

Comme les exigences sont la base pour le développement de système, toute erreur ou exigence manquante se propagera aux autres processus de développements dans le projet. Il est important de noter que les défauts résultant d'exigences de faible qualité sont plus onéreux à corriger dans les phases ultérieures du projet comparativement aux autres types de défaut. De plus, plus les défauts sont détectés tardivement, plus le coût de leur correction est élevé.

Par conséquent, la vérification (« produisons-nous le produit correctement ? ») et la validation (« produisons-nous le bon produit ? ») des exigences sont des activités nécessaires. La validation et la vérification devraient être effectuées continuellement durant le développement de la solution afin d'assurer que le produit en développement atteint le critère de qualité et satisfait les besoins des acteurs. La meilleure pratique est de planifier et d'effectuer la validation et la vérification des exigences dès les phases amont du développement de la solution – idéalement à partir de la phase d'élicitation des exigences. Les techniques communes pour la validation et la vérification des exigences incluent différents types de revue, du prototypage ou des présentations des solutions proposées ou des exigences aux parties prenantes, dans le but de recevoir du retour d'information (commentaires, remarques).

Les activités de validation et de vérification devraient aussi inclure l'assurance que les exigences et/ou les spécifications d'exigences/solution se conforment aux standards de la société (modèles) et qu'elles sont documentées et testées vis-à-vis des critères qualité des exigences (cf. section 1.1.4 Qualité des exigences). Il est aussi important de valider les modèles développés durant l'analyse des exigences et les activités de spécification.

Puisque les exigences sont la base pour le développement de solution et le test, leur qualité est cruciale pour le succès du projet. Des exigences claires, complètes, cohérentes et testables réduisent le risque d'échec du produit (ou même plus du projet), car elles permettent de tester avec soin. Il est ainsi recommandé d'impliquer les tests dans les revues des exigences car ils peuvent, de façon significative, aider à l'amélioration de la qualité des exigences et de la spécification des exigences en identifiant les points faibles et les défauts possibles.

La testabilité des exigences est supportée par un critère d'acceptation. Le critère d'acceptation décrit le critère qui doit être atteint pour approuver la solution et il devrait être agréé de part et d'autre – le fournisseur et le client – avant de démarrer le projet. La meilleure pratique est de définir le critère d'acceptation dans la documentation contractuelle. Chaque exigence de haut-niveau devrait avoir au moins un critère d'acceptation et chaque critère doit être mesurable par des

moyens réalistes et agréés. Un tel critère crée souvent la base pour le plan qualité du projet et l'activité de test d'acceptation.

[Pour les organismes de formation : fournir des exemples de critère d'acceptation pour des échantillons d'exigences.]

6 L'Ingénierie des Exigences dans les Modèles	100 minutes
--	--------------------

Termes

CCMI, approche Agile, approche itérative, modèle de maturité, modèle de processus, backlog de produit, cycle de vie de produit, propriétaire de produit, processus rationnel unifié (RUP), approche séquentielle, SPICE

Objectifs d'apprentissage pour le niveau fondation de l'Ingénierie des Exigences

Les objectifs identifient ce que vous serez capables de démontrer suivant l'achèvement de chaque module.

6.1 Modèles et approches de développement et de maintenance

Modèles et approches de développement et de maintenance

- LO-6.1.1 Résumer les caractéristiques communes d'un modèle de cycle de vie de produit (K2)
- LO-6.1.2 Reconnaître les similarités et les différences dans les processus d'Ingénierie des Exigences entre les modèles communs décrivant le développement de produit ou le processus de maintenance (K2)

6.2 Modèles de maturité

- LO-6.2.1 Comprendre comment l'Ingénierie des Exigences fait partie des modèles communs de maturité (K2)

<h2>6.1 Approches de Développement et de Maintenance des modèles</h2>	<h2>60 minutes</h2>
---	---------------------

La structure du processus d’Ingénierie des Exigences dépend du modèle de développement utilisé. Le processus générique d’Ingénierie des Exigences devrait être adapté afin de correspondre aux besoins spécifiques et à l’environnement de développement.

Le but de ce chapitre est de fournir des exemples d’occurrence de différentes activités d’Ingénierie des Exigences dans des types standards de modèles de développement.

Les modèles de processus sont des descriptions de processus de développement indépendants des méthodes. Les modèles de processus décrivent habituellement des rôles, des activités, des phases et des documents, et ils fournissent un format standard pour planifier, organiser et exécuter un projet.

Un cycle de vie de produit (PLC : *Product Life Cycle*) générique définit plusieurs phases de développement de produit où les phases basiques sont : la planification, le développement, la maintenance et la fin de vie. La phase de développement est souvent divisée en plusieurs phases, souvent effectuées itérativement (Figure 2 – Cycle de vie de produit basique (PLC)).

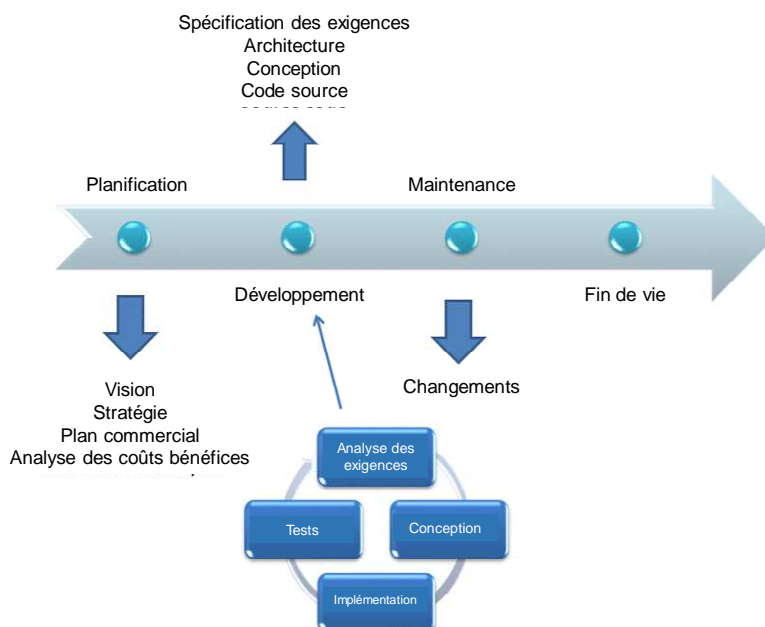


Figure 2 - Cycle de vie de produit basique (PLC)

Ce PLC générique est une base pour différents modèles de développement.

En général, les modèles de développement peuvent être classés en approches traditionnelles et en approches Agile. Les approches traditionnelles peuvent être divisées en modèles séquentiels, modèles itératifs et modèles incrémentaux (les deux derniers sont souvent combinés).

Le modèle de développement le plus simple – le modèle en cascade – peut être vu comme une représentation linéaire du processus générique d'Ingénierie des Exigences, basé sur l'hypothèse qu'il est possible de collecter, définir et d'analyser toutes les exigences dans une phase amont du développement et qu'elles ne seront pas sujettes à changement. La principale faiblesse de ce modèle de processus est sa faible aptitude aux changements.

Le modèle général en V est issu d'un développement ultérieur du modèle en cascade original. La principale amélioration implémentée dans le modèle en V est la distinction entre les étapes de développement et les étapes de test, où chaque étape de développement a une étape de test correspondante. Le modèle peut être vu comme une décomposition d'exigences où chaque niveau a un niveau de test associé.

Ce modèle permet, d'une part, une mise en œuvre optimum de l'Ingénierie des Exigences dans les phases de développement et, d'autre part, fournit une bonne base pour la gestion des exigences (traçabilité, gestion des changements). Toutefois, le développement des exigences dans le modèle en V peut être difficile puisqu'il est souvent impossible d'assurer que toutes les exigences importantes sont identifiées dans les phases amont du projet. Très souvent, le périmètre initial du produit augmente en raison d'exigences venant des différents acteurs et des systèmes environnants ou environnements, dans les phases tardives. Cela peut engendrer de vastes remaniements et de gestion du changement

Le modèle en V peut être considéré comme un des meilleurs modèles à la fois pour le développement et pour la gestion des exigences compte tenu des périmètres bien définis et stables.

[Pour les organismes de formation : représentation graphique du modèle général en V ; description du processus d'Ingénierie des Exigences dans le modèle général en V.]

Dans le cas de solutions plus complexes ou instables, il est recommandé d'utiliser des modèles itératifs ou incrémentaux. L'approche itérative/incrémentale est construite sur le fait que les exigences sont souvent difficiles à spécifier, spécialement dans les systèmes commerciaux. Les techniques de développement itératives et incrémentales supposent coopération et partenariat entre les développeurs et les utilisateurs. Les exigences sont souvent définies en termes d'objectif métier seulement – ces objectifs métier sont alors élaborés dans des itérations ultérieures.

Le développement incrémental suppose que la liste des exigences soit divisée en parties plus petites et que chaque partie est développée en étapes ou incréments. L'idée est que chaque partie

peut être implémentée et, dans le meilleur des cas, prend corps de sorte que le temps de mise sur le marché est court. Il est aussi possible d'avoir un retour d'expérience immédiat sur les incréments délivrés, si bien que les incréments en suspens ou non encore développés peuvent être améliorés. De cette façon le risque peut être réduit (spécialement en comparaison à une stratégie « d'un seul coup (one go) »). Même si le projet est terminé avant achèvement, les acteurs devraient obtenir des bénéfices des incréments déjà terminés.

L'approche itérative est particulièrement utile pour les gros projets avec un vaste nombre d'exigences. La principale idée de l'approche itérative est de développer une solution par l'intermédiaire de cycles répétés (qui peut être considéré comme un mini-PLC) dans des portions de temps plus petites. Le but d'une telle organisation est de rendre le projet plus contrôlable et d'être capable de superviser et d'aborder les risques. Un exemple de modèle itératif est RUP®, Rational Unified Process, un modèle de processus d'IBM Rational.

RUP est un modèle pour le développement de produit logiciel dans un contexte orienté objet. Il organise le projet autour de 4 principales phases : commencement, élaboration, construction et transition. Chacune de ses phases a ses propres buts et produits, définis par le cadre de travail RUP. RUP est un modèle itératif où les contenus des itérations dans une phase spécifique sont choisis en fonction des risques des éléments implémentés. Le nombre d'itérations dans une phase spécifique peut différer et il est ajusté aux besoins spécifiques du projet ou du produit. Le travail d'Ingénierie des Exigences est généralement couvert par les exigences et les disciplines d'analyse et de conception définies par le cadre de travail RUP.

[Pour les organismes de formation : approfondissement de RUP® avec présentation graphique ; étude approfondie de la discipline des exigences.]

Tous les modèles de processus décrits ci-dessus appartiennent aux approches dites « traditionnelles ». Le défi le plus courant des approches traditionnelles est de prendre en compte les changements fréquents. La plupart des modèles traditionnels ne prennent pas efficacement en compte les exigences changeantes. En tant que solution à ce problème, les méthodes dites « Agile » sont apparues. Les approches Agile sont basées sur le Manifeste Agile qui comporte un jeu de propositions et de principes utilisés en environnement Agile. Les plus importants principes Agile sont [Manifeste Agile] :

- Satisfaire le client par des livraisons rapides et continues de logiciel qui apportent de la valeur
- Accueillir les changements dans les exigences, même tard dans le développement
- Livrer fréquemment des logiciels exploitables (semaines plutôt que mois)
- Simplicité – l'art de maximiser la quantité de travail non-effectué – est essentiel

- Client et développeurs travaillent chaque jour ensemble tout au long du projet

Les implications de ces principes pour l'Ingénierie des Exigences sont tout à fait significatives. En environnements Agile, les exigences sont souvent communiquées et suivies par l'intermédiaire de mécanismes tels que les « Product Backlogs », « User Stories », « Story Cards » ou « Screen mock-up ». Les engagements sur les exigences sont soit effectués collectivement par l'équipe ou par un chef d'équipe habilité. La traçabilité et la cohérence des exigences et les produits du travail doivent être gérés par les mécanismes déjà mentionnés (Product Backlogs, User Stories, story cards, et screen mock-ups) et durant les activités de début et de fin d'itération telles que les « rétrospectives » et les « demo days ». Le processus de gestion des exigences n'est donc pas aussi formel que dans les modèles traditionnels de développement et de maintenance.

Le processus de développement des exigences est étroitement lié à la définition de la description et de l'analyse spécifiques des « User Stories » ou scénarios. Quantité d'activités normalement effectuées sous le développement des exigences ne sont pas décrites dans les modèles Agile mais sont supposées être effectuées par l'équipe de développement Agile sous la responsabilité d'un « Product Owner ».

Dans les environnements Agile, les besoins et les idées des clients sont élicités, élaborés, analysés et validés itérativement. Les exigences peuvent être rassemblées dans des « Backlogs » et spécifiées sous la forme de « User Stories », scénarios ou cas d'utilisation. La sélection des exigences qui seront adressées dans une itération donnée s'effectue sur la base de l'évaluation des risques ainsi que sur les priorités associées aux exigences présentes dans le « Product Backlog ». La sélection des exigences (ainsi que des artefacts) qui seront documentés est lié au besoin de coordination (parmi les membres de l'équipe, entre les équipes et pour les itération suivantes) et le risque de perdre la connaissance acquise. Quand le client est dans l'équipe, il y a aussi nécessité de séparer la documentation client de la documentation liée au produit pour permettre l'exploration de différentes solutions. A mesure que la solution émerge, les responsabilités liées aux exigences sont réparties entre les équipes appropriées.

Des exemples d'approches Agile sont : « Extreme Programming » et « Scrum ».

« Extreme Programming » est une méthodologie de développement logiciel développée par Kent Beck et al. dans le but de répondre aux changements des exigences du client. Il préconise des livraisons fréquentes de logiciel au client dans des cycles de développement courts (time boxing) dont les buts sont d'améliorer la productivité et de fournir des points de contrôle où des nouvelles exigences peuvent être adoptées.

Des caractéristiques de « Extreme Programming » incluent :

- Eviter de programmer des fonctionnalités tant qu'elles ne répondent pas un besoin effectif.
- Attendre des changements dans les exigences du client qui permettront au cours du temps de mieux comprendre le problème.
- Une communication fréquente avec le client et parmi les programmeurs.
- Un renoncement complet à déterminer les exigences (pas de séparation des phases d'exigences et de début des développements ; développement des exigences, raffinement et découverte des exigences font parties du développement logiciel effectif (programmation)).

Scrum est un framework Agile contenant un ensemble de pratiques et de rôles prédéfinis. Une des caractéristiques majeures de Scrum est de diviser les développements en « Sprints » (typiquement une période de 2 à 4 semaines). Durant chaque « Sprint », l'équipe crée ce qui s'appelle « un incrément de produit potentiellement livrable ».

Les projets Scrum gèrent les exigences par l'intermédiaire des « backlogs ». Il y a deux types de backlogs :

- Le « Backlog » de produit – une liste de haut-niveau maintenue durant tout le projet. Il agrège les exigences sous la forme de descriptions générales de toutes les potentielles caractéristiques, priorisées par valeur métier. Le « backlog » de produit est la propriété du « Product Owner ».
- Le « Sprint Backlog » – la liste du travail devant être pris en charge par l'équipe durant le prochain « Sprint ». Les caractéristiques sont subdivisées en tâches lesquelles, en tant que bonne pratique, doivent normalement représenter entre 4 et 16 heures de travail. Le « Sprint backlog » est la propriété de l'équipe.

Le principal impact de Scrum sur l'Ingénierie des Exigences porte sur les spécifications des exigences ne sont pas achevées et validées avant le commencement du développement.

Le « Product Owner » et l'équipe s'accordent sur les caractéristiques du backlog de produit devant être inclus dans le prochain sprint basé sur les priorités métier et sur l'effort de travail requis. Les exigences d'utilisateur sont formulées par le « Product Owner » sous la forme de « User Stories » qui contiennent de l'information sur « qui, quoi, pourquoi » d'une exigence. Au début du « Sprint », les caractéristiques sélectionnées sont subdivisées en tâches du « Sprint Backlog », et elles sont alors développées.

L'implication des « Product Owner », par exemple en visualisant des présentations des fonctions implémentées du logiciel, aide à clarifier les exigences pour l'équipe et les « Product Owner » eux-mêmes.

[Pour les organismes de formation : donner des exemples de User Stories, et de produit et d'items de « Sprint Backlog » correspondants.]

6.2 Modèles de maturité

40 minutes

Le processus d'Ingénierie des Exigences peut être amélioré tout autant qu'un autre processus défini comme une étape de développement de produit/solution ou de maintenance. Les améliorations peuvent être appliquées à chacune des activités d'Ingénierie des Exigences. Par exemple, l'élicitation des Exigences pourrait être améliorée pour une collecte d'exigences Métier plus efficace. Pour ce faire, l'organisation devra proposer des techniques de collecte d'exigences afin qu'elles soient élucidées plus rapidement, qu'elles soient complètes et qu'elles soient acceptées par la majorité des parties prenantes. Ces techniques incluent les entretiens, le brainstorming, le prototypage initial, l'utilisation de Personas et les scénarios.

Une méthode commune d'amélioration du processus d'Ingénierie des Exigences est d'employer des modèles de maturité. Ces modèles exigent l'évaluation du niveau actuel de maturité du processus et l'identification de points d'amélioration. Les modèles de maturité utilisent souvent des niveaux de maturité. Les niveaux de maturité sont utilisés pour l'identification, l'évaluation et l'amélioration de la maturité de processus.

Les exemples des modèles de maturité pouvant être appliqués au processus d'Ingénierie des Exigences sont ISO/IEC 15504 (SPICE - Software Process Improvement and Capability Determination) ou le Modèle de Maturité de Capacité Intégré (CMMI). Ces deux modèles définissent cinq niveaux de maturité pour des points ou des processus spécifiques et permettent de comparer la maturité à travers des organisations différentes.

ISO/IEC 15504 (SPICE - Software Process Improvement and Capability Determination), est un ensemble de normes techniques pour le processus de développement logiciel et les fonctions de gestion d'entreprise liées. SPICE peut être utilisée comme un moyen pour traiter l'amélioration et/ou la détermination de capacité (par exemple, l'évaluation de la capacité de processus du fournisseur). SPICE définit cinq niveaux de maturité pour les processus suivants : Client/Fournisseur, Ingénierie, Support, Direction et Organisation. Ces cinq niveaux sont :

- 1 Processus exécuté
- 2 Processus géré
- 3 Processus établi
- 4 Processus prévisible
- 5 Processus en optimisation

Les domaines de Processus clés liés à l'Ingénierie des Exigences dans le modèle SPICE sont :

- Préparation de l'acquisition BP1-3
- Elicitation des exigences BP1-6
- Analyse des exigences système BP1-6
- Analyse des exigences de solution BP1-6

Le Capability Maturity Model Integrated (CMMI) définit cinq niveaux de maturité pour le développement, des services et l'acquisition :

1. Initial
2. Discipliné
3. Ajusté
4. Géré quantitativement
5. Optimisation

Les domaines de processus clés liés à l'Ingénierie des Exigences dans le modèle CMMI sont :

- REQM : Gestion des exigences
- RD : Développement des exigences

Le processus d'Ingénierie des Exigences peut être lui-même considéré comme un moyen d'amélioration de processus de développement en admettant qu'une Ingénierie des Exigences efficace augmente les efforts de mise en œuvre en éliminant le temps gaspillé de clarification et d'explication des exigences, en réduisant le nombre de défauts causés par des erreurs de spécification d'exigences et/ou des malentendus dans la documentation d'exigences en raison de leur mauvaise qualité. Un bon processus d'Ingénierie des Exigences augmente la qualité des tests en fournissant des exigences bien décrites, précises et testables et augmente l'efficacité et la fiabilité des tests d'acceptation en admettant que le test soit basé normalement sur les exigences.

[Pour les organismes de formation : démontrer les exigences typiques d'un processus d'Ingénierie des Exigences dans des modèles de maturité de processus.]

7 Support d'Outils

80 minutes

Termes

Outil de gestion de Changement, Outil de gestion d'anomalies, Outil de modélisation, Projet Pilote, Outil de Gestion de projet, Preuve de Concept, Outil de prototypage, Outil d'Elicitation des Exigences, Outil de gestion des Exigences, Catégories d'Outils

Objectifs d'apprentissage pour Niveau Fondation d'Ingénierie des Exigences.

Les objectifs identifient ce que vous pourrez démontrer après la réalisation de chaque module.

7.1 Avantages des outils

- LO-7.1.1 Expliquer les avantages d'utiliser des outils pour soutenir les différentes activités de l'Ingénierie des Exigences (K2)

7.2 Catégories des outils

- LO-7.2.1 Comprendre le but et l'utilisation de chacune des catégories d'outils soutenant l'Ingénierie des Exigences (K2)

7.3 Choix des outils

- LO-7.3.1 Expliquer les facteurs importants devant être considérés pour choisir un outil soutenant l'Ingénierie des Exigences (K2)
- LO-7.3.2 Expliquer le processus d'introduction d'un outil dans une organisation (K2)

7.1 Avantage des Outils

30 minutes

Il existe différents outils supportant les activités spécifiques de l'Ingénierie des Exigences.

Une application commune à un certain nombre d'outil est de faciliter la sauvegarde et l'administration des exigences. La plupart des outils de gestion des exigences fournissent un Référentiel commun pour les exigences, où les exigences peuvent être organisées et la traçabilité maintenue. Il est ainsi possible de garder des documents complexes et changeants cohérents et à jour. Ces outils peuvent automatiser quelques activités manuelles et peuvent fournir différentes statistiques et rapports. Les outils de modélisation permettent de modéliser les exigences dans des notations de modélisation spécifiques, comme UML ou BPMN. Très souvent les outils de modélisation peuvent générer le document de spécifications à partir des modèles.

En général, les outils peuvent supporter les activités de l'Ingénierie des Exigences suivantes :

- Elicitation des Exigences et sauvegarde
- Définition et gestion de la traçabilité des exigences
- Modélisation des Exigences et de la solution (incluant le prototypage et la modélisation d'activité)
- Documentation des Exigences (création de la spécification des exigences)
- Reporting
- Gestion de Configuration et du Changement

Quelques avantages liés à l'utilisation d'outils :

- S'assurer que toutes les exigences sont stockées à un endroit unique et accessible par toutes les parties prenantes impliquées
- Soutenir la traçabilité des exigences (par exemple, jusqu'aux cas de tests) qui peut être utilisé pour vérifier la pertinence de la couverture des exigences
- Gérer les changements dans les exigences d'une manière simple qui inclut le contrôle de configuration
- Améliorer la qualité des spécifications des exigences en forçant l'utilisation de modèles de document définis et la notation de modélisation
- Économiser du temps en automatisant certaines activités (telles que la génération des spécifications complètes de l'outil).
- Fournir de l'information pour la prise de décision (par exemple, les paramètres indiquant le nombre d'exigences mises en œuvre).

7.2 Catégories d'Outils

20 minutes

Les outils supportant les activités d'Ingénierie des Exigences peuvent être classés comme suit :

- Outils de Gestion des Exigences
- Outils d'Elicitation des Exigences
- Outils de modélisation des Exigences
- Outils de gestion des anomalies
- Outil de gestion de configuration et du changement
- Outil de gestion de projet

Beaucoup d'outils peuvent être utilisés pour plus d'une activité, par exemple les outils de modélisation des exigences peuvent offrir un Référentiel des exigences avec des facilités de gestion de configuration et de changement, proposer différentes notations de modélisation, la génération de documentation et des statistiques.

[Pour les organismes de formation : présenter un exemple d'un outil supportant au moins deux activités d'Ingénierie des Exigences et fournir des exemples concrets d'application.]

7.3 Sélection des Outils

30 minutes

La sélection d'un outil doit se faire avant que le produit soit conçu pour éviter les problèmes qui peuvent survenir lors d'un changement d'outils à mi-projet.

Le coût d'achat des outils varie considérablement. Les outils commerciaux peuvent être très chers, alors que des outils open source sont gratuits. Le choix d'un outil doit être fait très soigneusement. Avant de choisir un outil, une analyse devra être menée. L'analyse pourra porter sur les questions suivantes :

- Évaluation de la maturité organisationnelle, les forces, les faiblesses et l'identification des possibilités d'amélioration du processus d'Ingénierie des Exigences au travers des outils
- Exigences concernant les caractéristiques nécessaires de l'outil
- Une preuve de concept (POC), en utilisant un outil de test au cours de la phase d'évaluation dans le but de déterminer si l'outil fonctionne efficacement lors du processus d'Ingénierie des Exigences donnée et dans l'infrastructure actuelle et d'identifier les changements potentiels nécessaires de l'infrastructure existante
- Estimation du retour sur investissement basé sur une analyse concrète de rentabilité qui peut inclure l'analyse du coût de l'outil pour un projet spécifique, tous les projets, une partie des projets
- Intégration possible entre un outil d'Ingénierie des Exigences et d'autres outils nécessaires (par exemple, l'outil de gestion des anomalies, l'outil de gestion de projet)
- Le besoin de communication entre un outil d'Ingénierie des Exigences et un outil utilisé par une organisation cliente (dans certains cas, les clients procèdent à leurs propres analyses des besoins et les résultats seraient ensuite migrés vers l'environnement du fournisseur)
- Facilité d'utilisation et d'apprentissage (coût potentiel de la formation), la disponibilité de l'aide en ligne, manuels, tutoriels, et autres formes de support

L'introduction de l'outil sélectionné dans une organisation doit commencer par un projet pilote. Les objectifs communs d'un projet pilote peuvent être les suivants :

- Apprendre plus de détails sur l'outil
- Évaluer la façon dont l'outil s'intègre avec les processus et les pratiques existantes, et déterminer ce qui devrait changer
- Décider les standards d'utilisation, de gestion, de stockage et de maintien de l'outil et les avantages d'Ingénierie des Exigences

Les projets pilotes réduisent le risque d'introduction d'outils qui, si celle-ci est effectuée à la hâte et sans analyse appropriée, peut entraîner des coûts élevés. Voici des exemples de résultats coûteux de mise en place d'outil :

- Sélectionner un outil qui ne répond pas aux exigences des parties prenantes et qui ne répond pas à son objectif
- Acheter un outil coûteux qui n'est utilisé que sur un seul projet ou acheter un outil coûteux alors qu'il existe des outils open source avec des fonctionnalités similaires
- Formation requise pour un outil n'ayant pas de système d'aide suffisante (en particulier lorsque que seule la fonctionnalité de base est nécessaire de l'outil)
- Extension de l'outil sélectionné avec des fonctions supplémentaires non prises en charge par l'outil (alors qu'il y avait d'autres outils avec une telle fonctionnalité)
- Intégration de l'outil avec d'autres outils utilisés dans l'organisation

En général, les facteurs de réussite pour un déploiement d'outil sont les suivantes :

- Adapter et améliorer les processus pour s'adapter à l'utilisation de l'outil
- Déployer l'outil dans l'organisation progressivement
- Assurer la formation et le coaching / mentorat pour les nouveaux utilisateurs, y compris la définition des lignes directrices d'utilisation
- Implémenter un moyen de recueillir des informations sur l'utilisation de l'outil à partir de l'utilisation réelle (y compris les mesures et les leçons apprises)
- Surveiller l'utilisation et les avantages de l'outil

8 Tables / Illustrations

Tables

Table 1 - Vue d'ensemble des activités de l'Ingénierie des Exigences, entrées et sorties.....	37
Table 2 - Techniques d'Elicitation des Exigences	75
Table 3 - Exemple de méthodes d'analyses et diagrammes	85
Table 4 - Les diagrammes UML et leurs applications	87

Figures

Figure 1 - Contexte de l'Ingénierie des Exigences	28
Figure 2 - Cycle de vie de produit basique (PLC)	97

9 Standards

IEEE 610.12-1990 IEEE Standard Glossary of Software Engineering Terminology

IEEE 830-1998 IEEE Recommended Practice for Software Requirements Specifications

IEEE 1233-1998 IEEE Guide for Developing System Requirements Specifications

IEEE 1362-1998 IEEE Guide for Information Technology-System Definition – Concept of Operations (ConOps) Document

IEEE 830, IEEE 1233 and IEEE 1362 are replaced by ISO/IEC/IEEE 29148:2011

ISO 9000 Quality management

ISO 12207 Systems and software engineering — Software life cycle processes

ISO 15288 System Life Cycle Processes

ISO 15504 Information technology — Process assessment, also known as SPICE (Software Process Improvement and Capability Determination),

ISO 31000 Risk Management - Principles and Guidelines on Implementation

ISO/IEC 25000 (previously ISO/IEC 9126 Software engineering — Product quality)

ISO/IEC 14598-1:1999 Information technology -- Software product evaluation

SWEBOK - The Guide to the Software Engineering Body of Knowledge :

<http://www.computer.org/portal/web/swebok/home>

SEBOK – Systems Engineering Body of Knowledge : <http://www.sebokwiki.org/>

10 Livres et publications

Cockburn, A.: *Writing Effective Use Cases*. Amsterdam, 2000

Cohn M.: *Estimating With Use Case Points*. Fall 2005 issue of Methods & Tools

Cohn M.: *User Stories Applied: For Agile Software Development*. Addison-Wesley Professional, 2004

Cotterell, M. and Hughes, B.: *Software Project Management*. International Thomson Publishing, 1995

Davis, A. M.: *Just Enough Requirements Management. Where Software Development Meets Marketing*. Dorset House, 2005, ISBN 0932633641

DeMarco, T.: *Controlling Software Projects : Management, Measurement and Estimates*. Prentice Hall, 1986

Doran, G. T.: *There's a S.M.A.R.T. way to write management's goals and objectives*. Management Review, Volume 70, Issue 11, 1981, (AMA FORUM), pp. 35-36.

Evans, E. J.: *Domain-Driven Design : Tackling Complexity in the Heart of Software*. Amsterdam, 2003

Graham, D. et al: *Foundations of Software Testing*. London, 2007

Gilb, T.; Graham, D.: *Software Inspection*. Reading, MA, 1993

Gilb, T.: *What's Wrong with Requirements Specification*. See : www.gilb.com

Hull, E. et al: *Requirements Engineering*. Oxford, 2005

IBAQB: *Certified Business Analyst, Foundation Level Syllabus*, version 2011

IBUQ: *Certified Professional for Usability Engineering, Foundation Level Syllabus*, version 2011

ISTQB: *ISTQB Glossary of Testing Terms v2.2*

ISTQB: *Certified Tester, Foundation Level Syllabus*, version 2011

Jacobson, I. et al.: *Object-Oriented Software Engineering. A Use Case Driven Approach*. Addison-Wesley, 1993

Lauesen, S.: *Software Requirements : Styles and Techniques*. London, 2002

Pfleeger, S.L. and J.M. Atlee : *Software Engineering: Theory and Practice*. Upper Saddle River, New Jersey, USA, Prentice Hall, 2006.

Project Management Institute : *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*. PMI, 2004

Robertson, S.; Robertson, J.: *Mastering the Requirements Process*. Harlow, 1999

Rupp, C.: *Requirements-Engineering und Management. Professionelle, Iterative Anforderungsanalyse in der Praxis*. Munich, 2007

Sharp H., Finkelstein A. and Galal G.: *Stakeholder Identification in the Requirements Engineering Process*. 1999

Sommerville, I.: *Requirements Engineering*. West Sussex, 2004

Sommerville, I.: *Software Engineering 8*. Harlow, 2007

Sommerville, I.; Sawyer, P.: *Requirements Engineering: A Good Practice Guide*. Chichester, 1997

Sommerville, I.; Kotonya, G.: *Requirements Engineering: Processes and Techniques*. Chichester, 1998

Thayer, R. H.; Dorfman, M.: *Software Requirements Engineering, 2nd edition*. Los Alamitos, CA, 1997

Wieggers, K.E.: *First Things First : Prioritizing Requirements*. Software Development, September, 1999

Wieggers, K. E.: *Software Requirements*. Redmond, 2005

Young, R. R.: *Effective Requirements Practices*. Addison-Wesley, 2001

11 Index

- Agile**, 41, 59, 80, 90, 92, 96, 98, 99, 100, 101, 112
- Analyse d'impact, 15, 42, 46, 47, 48, 50, 51, 53, 54, 55, 76, 101
- Analyse des Exigences, 34, 35, 40, 61, 62, 63, 71, 72, 78, 82, 83, 84, 88, 89, 91, 92, 94, 104
- Analyse des Points de Cas d'Utilisation, 80
- Analyse des risques, 36, 37, 48
- Analyse métier**, 27, 28, 29, 30, 34
- Apprentissage, 61, 67, 73
- Approche Itérative, 96, 98, 99
- Approche séquentielle, 96
- Assurance Qualité, 10, 12, 23, 37, 42, 57, 58
- Atelier, 61, 74
- Attributs qualité, 14, 17, 75
- Auto-enregistrement, 61, 67, 69
- Besoin métier, 9, 16, 18, 29, 35
- BPMN**, 61, 85, 88, 92, 106
- Brainstorming**, 61, 67, 75, 103
- But, 64, 65
- Cas d'utilisation, 61, 62, 67, 70, 77, 81, 84, 85, 86, 88, 90, 100
- Changement, 2, 18, 19, 22, 23, 28, 29, 30, 34, 36, 37, 38, 40, 42, 43, 44, 45, 46, 49, 50, 52, 53, 54, 55, 56, 57, 60, 63, 66, 72, 79, 83, 98, 99, 100, 101, 105, 106, 107, 108
- Client**, 9, 10, 16, 18, 19, 20, 21, 22, 24, 29, 32, 37, 38, 39, 41, 45, 46, 48, 55, 57, 61, 64, 65, 66, 69, 73, 76, 83, 84, 88, 89, 90, 94, 99, 100, 101, 103
- CMMI**, 22, 23, 26, 103, 104
- Comité de Contrôle du Changement**, 42, 43, 54, 55
- Contrainte, 14, 17, 24, 47
- Contrainte métier**, 14, 17
- Contrainte technique, 14, 17
- Contrat**, 17, 19, 45, 61, 65, 89
- Contrôle qualité, 10, 42, 57, 58
- Critère d'acceptation**, 61, 62, 65, 89, 94, 95
- Criticité**, 14, 19
- Cycle de vie du produit, 44, 52
- Demande de changement**, 42, 43, 53, 54
- Développement des Exigences, 11, 15, 23, 32, 34, 35, 44, 50, 61, 63, 64, 89, 98, 100, 101, 104
- Développeur des Exigences, 32, 39, 40
- Documentation des exigences**, 83
- Elicitation des Exigences, 12, 25, 40, 61, 63, 64, 66, 67, 75, 78, 84, 94, 103, 104, 105, 106, 107
- Engagement**, 14, 17, 19, 20, 100
- Entretien, 61, 67, 73, 103

Evaluation des risques, 48, 100
 Exigence, 14, 16, 18, 20
 Exigence de composant, 35, 83
 Exigence Fonctionnelle, 14, 17, 23, 51, 59, 75, 76, 84, 86, 87, 90
Exigence métier, 7, 14, 18, 19, 35, 46, 51, 77, 83, 89, 91, 92
 Exigence produit, 14, 17, 18, 22
 Exigences de systèmes/solution, 78, 83
Formelle, 83, 92, 93
 Fournisseur, 7, 8, 19, 20, 32, 37, 39, 41, 46, 47, 65, 66, 83, 84, 89, 94, 103, 108
Gestion de Configuration, 30, 34, 42, 43, 44, 52, 53, 106, 107
 Gestion de projet, 23, 30, 55, 57, 105, 107, 108
 Gestion des Exigences, 11, 14, 23, 30, 32, 34, 36, 39, 42, 44, 46, 52, 57, 82, 98, 100, 104, 105, 106, 107
 Gestion des risques, 30, 42, 46, 47
Gestion du Changement, 23, 42, 43, 44, 45, 52, 53, 54, 57, 72, 98
 Gestionnaire des Exigences, 32, 39
 Identification du risque, 47, 48
 Identification sur la base de documents existants, 61, 67, 71
 IEEE 1233, 20, 25, 91, 111
 IEEE 1362, 25, 91, 111
 IEEE 610, 16, 25, 52, 111
 IEEE 830, 19, 25, 51, 90, 111
 IHM, 10, 51, 61, 88
 Ingénierie des Exigences, 7, 8, 9, 10, 11, 12, 14, 15, 16, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 32, 33, 34, 37, 38, 39, 40, 42, 43, 45, 46, 52, 57, 58, 59, 61, 65, 72, 79, 92, 96, 97, 98, 99, 100, 101, 103, 104, 105, 106, 107, 108
 Ingénieur des Exigences, 7, 11, 48, 70, 73
 ISO 12207, 26, 111
 ISO 15288, 26, 111
 ISO 15504, 26, 103, 111
 ISO 9000, 25, 57, 111
 ISO 9126, 25, 76, 111
 ISO/IEC 25000, 25, 76, 111
 ISO/IEC/IEEE 29148, 19, 20, 25, 51, 111
Méthode Delphi, 61, 80
Modèle conceptuel, 2, 61, 78, 84
 Modèle de Kano, 20
 Modèle de la Solution, 84
 Modèle de processus, 96, 98, 99
 Modèles de maturité, 33, 96, 103, 104
 Non-fonctionnelle, 14, 70, 76, 84, 88, 93
 Non-formelle, 93
Observation terrain, 61, 67, 73
 Outils, 10, 34, 35, 37, 105, 106, 107, 108
 Parties-prenantes, 32, 35, 36, 41, 45, 47
 Planning Poker, 61, 80
 Point de Fonction, 80
 Preuve de concept (POC), 108

Niveau fondation

Priorisation, 12, 40, 61, 62, 78, 82
 Priorité, 14, 19, 20, 21, 59, 65, 74, 82, 91, 100, 101
 Problème métier, 12, 14, 16, 28, 75, 84, 89, 92
 Processus Générique d'Ingénierie des Exigences, 32, 34, 97, 98
 Product Backlog, 74, 100
 Product Owner, 100, 101
 Produit, 16, 17, 24, 46, 47
 Projet pilote, 105, 108, 109
 Prototypes de scenario, 88
 Prototypes horizontaux, 88
 Prototypes verticaux, 88
 Questionnaire, 61, 67, 80
 Rational Unified Process, 77, 96, 99
Référentiel, 36, 42, 106, 107
 Représentant du client, 10, 55, 61, 69
 Réutilisation, 50, 61, 67, 72
 Risque, 9, 19, 20, 23, 30, 42, 45, 46, 47, 48, 54, 55, 57, 59, 60, 63, 66, 83, 94, 99, 100, 109
 Risque du projet, 47
 Risques produit, 42, 47
 S.M.A.R.T, 61, 65, 66, 112
 SEBOK, 25, 111
 Solution, 16, 17, 19, 21, 23, 46, 58, 78, 79, 82, 84
 Spécification, 17, 18, 21, 22, 25, 30, 34, 35, 36, 37, 52, 58, 59, 61, 62, 63, 72, 77, 83, 84, 89, 90, 91, 92, 93, 94, 104, 106
 Spécification de la solution, 58
 Spécification des Exigences, 21, 22, 25, 30, 34, 35, 36, 37, 62, 63, 72, 77, 89, 90, 94, 106
 Spécification semi-formelle, 93
 Spécifications de solution, 21
 SPICE, 26, 96, 103, 111
 Sprint Backlog, 101, 102
 Suivi des Exigences, 100
 SWEBOK, 19, 23, 25, 111
 SysML, 85, 87
 Système, 25, 39
 Système, 10, 14, 16, 17, 18, 21, 22, 25, 26, 35, 36, 41, 47, 53, 55, 56, 57, 59, 66, 70, 73, 75, 76, 77, 80, 82, 83, 85, 86, 87, 89, 90, 91, 92, 93, 94, 104, 109
 Test, 18, 30, 43, 47, 59
 Testabilité, 58, 59, 60, 94, 104
 Traçabilité, 12, 18, 30, 34, 36, 37, 40, 42, 44, 46, 48, 49, 50, 51, 53, 54, 55, 57, 58, 76, 92, 98, 100, 106
 Traçabilité horizontale, 42, 50, 51
 Traçabilité verticale, 42, 51
 UML, 61, 62, 84, 85, 86, 87, 88, 92, 93, 106
 User Story, 61, 90
 Validation, 14, 15, 22, 23, 25, 34, 36, 38, 45, 50, 57, 58, 62, 63, 69, 78, 94

Validation des Exigences, 14, 15, 22, 23, 25,
34, 36, 38, 45, 50, 57, 58, 62, 63, 69, 78,
94

Vérification, 14, 15, 22, 23, 25, 34, 36, 45,
50, 57, 58, 62, 63, 78, 94

Vérification des Exigences, 14, 15, 22, 23,
25, 34, 36, 45, 50, 57, 58, 62, 63, 78, 94

Vision, 15, 45, 61, 64, 65, 66, 67, 91