



Ingénierie Logicielle

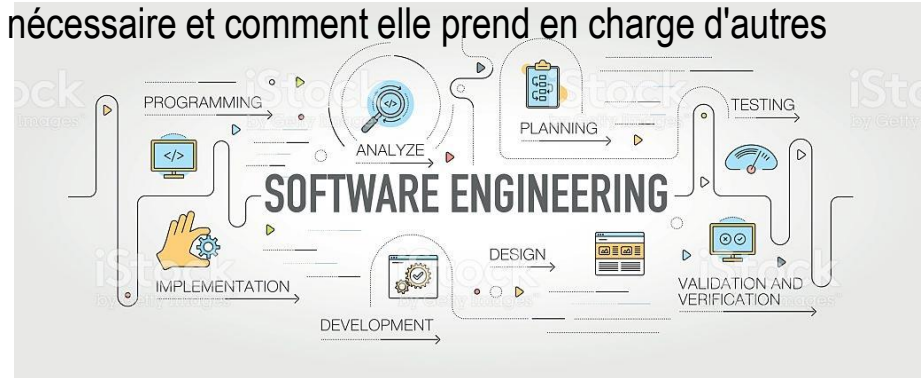
Analyse et Spécification des besoins

Prof. Ousmane SALL, Maître de Conférences CAMES en Informatique

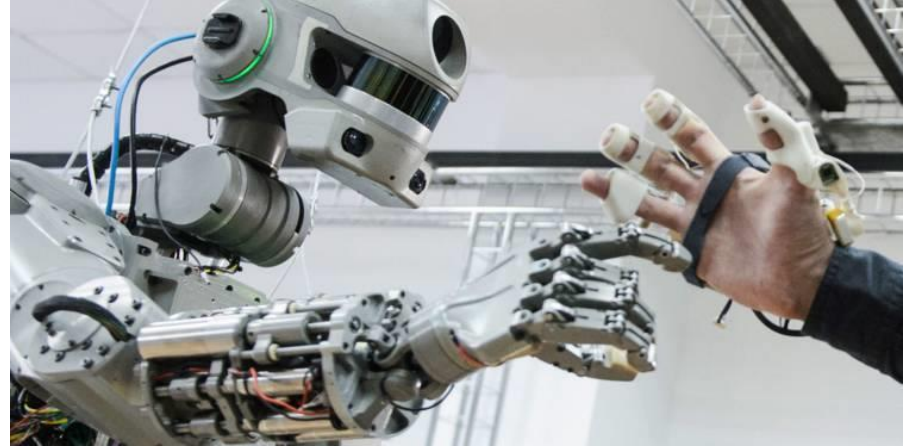


Objectifs de la séquence

- A l'issue de cette séquence vous serez capable de :
 - comprendre les concepts des besoins de l'utilisateur et du système et pourquoi ces besoins doivent être rédigés de différentes manières;
 - comprendre les différences entre les exigences logicielles fonctionnelles et non fonctionnelles;
 - comprendre comment les besoins peuvent être organisés dans un document des besoins de logiciel;
 - comprendre les principales activités d'ingénierie des besoins en matière de spécification, d'analyse et de validation, ainsi que les relations entre ces activités;
 - comprendre pourquoi la gestion des besoins est nécessaire et comment elle prend en charge d'autres activités d'ingénierie des besoins.



Les logiciels et la société



Génie logiciel

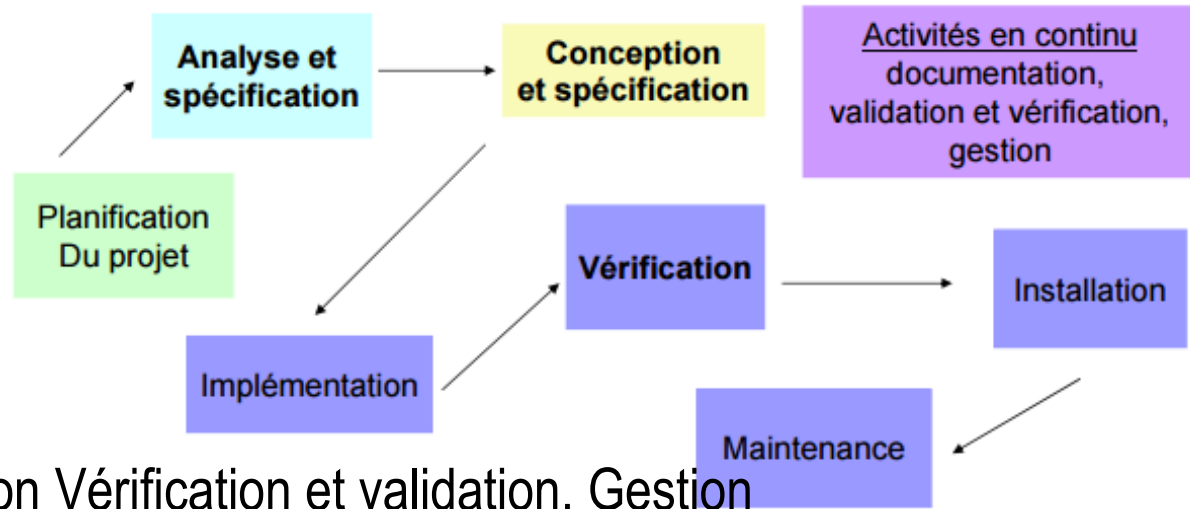
- Mettre en place des méthodes efficaces de management
- Créer de nouveaux outils et améliorer les outils existants
- Établir des normes de gestion, d'organisation, de communication
- Une discipline qui comprend:
 - Le processus de développement de logiciels
 - La méthodologie pour l'analyse, la conception, le développement, la vérification et la maintenance de logiciels
 - Des outils qui supportent le processus et la méthodologie

Génie logiciel / Ingénierie logicielle: Science qui s'intéresse aux méthodes de travail et aux bonnes pratiques de développement.

Cycle de vie du logiciel : Activités de développement (phases) ou Processus de Développement de Logiciel

- Planification du projet
- Analyse et spécification
- Conception
- Implémentation
- Vérification
- Installation
- Maintenance

Organisation de ces activités dans le temps (processus de développement).



- En continu: Documentation Vérification et validation, Gestion

Cycle de vie du logiciel: Analyse des besoins/Spécifications

- Aussi appelée "**Spécification**"
- Comprendre les besoins: **Quoi? Pourquoi?**
 - Étude du contexte: type de produit, intervenant, etc.
 - Comprendre les besoins et les objectifs: le produit devrait permettre de faciliter quoi?
 - Comprendre le processus existant: Comment procède le client avant (aide à comprendre attentes)
 - Étudier le marché: produits similaires, étudier les pt forts et limites des concurrents
- Étudier la faisabilité
- A l'issue de cette phase: **Autoriser ou pas le projet**

- Documents:
 1. Cahier des charges
 2. Document de spécification (analyse)
 3. Prototype
 4. Plan de test.

Cycle de vie du logiciel: Analyse des besoins

Phase difficile: Le client et les informaticiens ne parlent pas le même langage !

- Travaux sur la modélisation du besoins:
 - Requirements engineering
- Pas d'algorithme miracle:
 - Longues discussions avec le client !
- Outils
 - UML (Diagramme de contexte, cas d'utilisations, ...)

- Documents:
 1. Cahier des charges
 2. Document de spécification (analyse)
 3. Prototype
 4. plan de test.

Analyse et Spécifications des besoins: **Besoins**



Besoins

- Besoin («requirement») = exigence que le système devrait satisfaire
- Synonyme: exigences, caractéristiques, requis
- Exemples: Système de contrôle d'un ascenseur
 - B1. Le programme doit planifier les activités de l'ascenseur de façon efficace et raisonnable
 - B2. Le programme doit illuminer l'indicateur du panneau d'arrivée correspondant à l'étage où l'ascenseur arrive
 - B3. Au dernier (resp. premier) étage, le panneau d'appel ne contient qu'un seul bouton, soit celui pour descendre (resp. monter)



Nouveaux besoins

- Besoins ajoutés durant le processus de développement
- Par exemple
 - Un bogue est rapporté par un utilisateur
 - Un utilisateur demande une nouvelle fonctionnalité
 - Un programmeur propose une amélioration
 - Un gestionnaire désire rendre un produit concurrentiel

Identification des besoins

- Identifier les besoins correctement est difficile!
- Causes les plus répandues de l'échec de projets:
 - Contribution insuffisante du client
 - Spécification incomplète des besoins
 - Modification des besoins
- Difficultés:
 - Problèmes complexes, connaissance limitée du domaine, clients sans connaissances techniques, communication intensive

Le rôle de l'analyste



Client / utilisateurs

- Expertise, jargon du domaine Indécis, opinion changeant selon l'offre Besoins ambigus, éléments manquants



Analyste

- Doit devenir aussi informé du fonctionnement de l'entreprise que les utilisateurs
- Doit devenir l'expert



Développeur

- Schémas Langages formels Spécifications souvent incompréhensibles pour les non initiés.

Identification des besoins

- Collecte des informations
 - Identification des opérations et procédés administratifs
 - “Que faites-vous ?”
 - Réalisation des opérations
 - “Comment le faites-vous ?”
 - Identification des informations requises pour réaliser les opérations
 - “Quelles informations utilisez-vous ?”



Types de besoins

- Les besoins peuvent traduire des exigences concernant:
 - Les fonctionnalités
 - L'environnement physique
 - Les interfaces
 - Les humains (utilisateurs)
 - La documentation
 - Les données
 - Les ressources
 - La sécurité
 - L'assurance de la qualité

Classification des besoins

- **Besoins fonctionnels: comportement, caractéristiques, capacités**
 - ex: « Le système lit les fiches des employés et imprime des chèques de paie »
 - Les autres besoins sont non-fonctionnels (contraintes)
- **Besoins d'utilisation: facteurs humains, aide, documentation**
 - ex: « La police utilisée pour le texte doit être lisible d'une distance de 2 mètres »
 - ex: « Ne pas utiliser les couleurs associées aux formes communes de daltonisme »
- **Besoins de fiabilité: fréquence des défaillances, récupération**
 - ex: « Si une défaillance se produit dans le système, ... »
- **Besoins de performance:**
 - ex: « Le temps de réponse du système est inférieur à 1 seconde pour 90% des accès. »
 - ex: « Le système doit être en mesure de traiter 1Mo de données transactionnelles par seconde. »

Classification des besoins

- Besoins de support: adaptation, internationalisation, maintenance
 - ex: « Le système devra permettre des changements fréquents dans la topologie du réseau. »
 - ex: « Le système devra pouvoir incorporer plusieurs composants externes (inconnus) pour le calcul des impôts. »
- Besoins de d'implémentation:
 - ex: « Le système doit utiliser Linux et Java »
 - Souvent dû à des contraintes budgétaires ou à la disponibilité du personnel qualifié

Caractéristiques des besoins

- Corrects: ne contiennent pas d'erreurs
- Clairs, sans ambiguïtés, intelligibles: compréhensibles par les programmeurs sans avoir à supposer ou deviner
- Cohérents: non-conflictuels, possibles à satisfaire en même temps
- Réalistes: respectent ce qui peut être accompli avec la technologie courante
- Pertinents: nécessaires et non-redondants
 - les besoins superflus devraient être éliminés

Caractéristiques des besoins

- **Complets**
 - complétude interne: tous les termes et concepts introduits sont définis
 - complétude externe: Pour chaque état du système et chaque entrée possible, on doit définir le comportement du système, les contraintes et les produits attendus
 - Difficile à atteindre en pratique
- **Vérifiables: facilement mesurables / quantifiables de façon à déterminer si l'implémentation atteint l'objectif visé**
 - ex: immédiatement, rapidement vs. en moins de 1s
- **Traçables: bien classés, gérés, identifiés**
 - chaque fonction du système peut facilement être mise en lien avec les besoins qu'elle satisfait

Analyse et Spécifications des besoins: Processus d'analyse des besoins



Processus d'analyse des besoins

A. Expression des besoins



B. Spécification et modélisation des besoins



Recueillir l'information.
Définir les caractéristiques du système
Bâtir des prototypes pour la découverte

Prioriser les caractéristiques.
Produire et évaluer des solutions de rechange.
Examiner les recommandations

A // B ou A → B

Processus d'analyse

- Expression des besoins
 - Participants: analyste, client et utilisateurs.
 - Document: cahier des charges
 - Rédigé par: le client en collaboration avec l'analyste
 - En langue naturelle
 - Découpage: en paragraphes exprimant clairement les buts, les besoins et les contraintes
- Spécification et modélisation des besoins
 - Participants: analyste
 - Document: dossier d'analyse et de spécification
 - Rédigé par: l'analyste
 - Notation graphique ou textuelle rigoureuse
 - Découpage: modèles statique, fonctionnel et comportemental

Collecte des informations

- Méthodes traditionnelles
 - Entrevue avec clients, utilisateurs et experts du domaine
 - Entrevue formelle: préparée, agenda, questions ouvertes/fermées
 - Entrevue informelles: encourager le client à parler
 - Questionnaires
 - accompagnent ou préparent l'entrevue
 - Type de questions (fermées de préférence)
 - Question à choix multiple
 - Question avec "cote" de la réponse (ex. d'accord, pas d'accord, etc.)
 - Question avec "ordonnancement" des réponses

Collecte des informations

Questionnaire de RMO

Ce questionnaire est envoyé à tout le personnel du service des ventes par téléphone. Comme vous le savez, RMO est en train de développer un nouveau système de soutien des clients pour la prise des commandes et le service à la clientèle.

Ce questionnaire a pour but d'obtenir de l'information préliminaire pour nous aider à définir les spécifications du nouveau système. Des discussions de suivi seront organisées par la suite pour permettre à tous d'élaborer sur ces spécifications.

1^{ère} partie. Répondez à ces questions fondées sur un quart de travail de quatre heures.

1. Combien d'appels téléphoniques recevez-vous?
2. Combien d'appels téléphoniques faut-il pour passer une commande?
3. Combien d'appels ne sont que des questions sur les produits de RMO?
4. Évaluez combien de fois pendant votre quart de travail les clients vous demandent des articles en rupture de stock?
5. De ces demandes d'articles épuisés, quel pourcentage sera placé sur une liste des commandes en souffrance?
6. Combien de fois un client passe-t-il une commande à partir d'un catalogue périmé?
7. Combien de fois un client annule-t-il une commande au milieu de la conversation?
8. Combien de fois des commandes sont-elles refusées pour cause de mauvais crédit?

2^e partie. Sur l'échelle de 1 à 7 qui suit, encerclez le chiffre correspondant à votre accord ou votre désaccord avec l'énoncé.

Question	Fortement d'accord			Fortement en désaccord			
Je pourrais mieux faire mon travail si je pouvais donner des descriptions plus longues des produits aux clients à qui je parle.	1	2	3	4	5	6	7
Je pourrais mieux faire mon travail si j'avais accès à l'historique d'achat du client.	1	2	3	4	5	6	7
Je pourrais mieux servir les clients si j'avais accès à de l'information sur les accessoires adéquats pour les articles commandés.	1	2	3	4	5	6	7
Le temps de réponse de l'ordinateur est lent et me cause des difficultés lorsque je réponds aux demandes des clients.	1	2	3	4	5	6	7

3^e partie. Veuillez inscrire vos opinions et commentaires.

Identifiez brièvement les problèmes du système actuel que vous aimeriez voir résolus par un nouveau système.

Questionnaire

Questions fermées objectives

Questions fermées subjectives

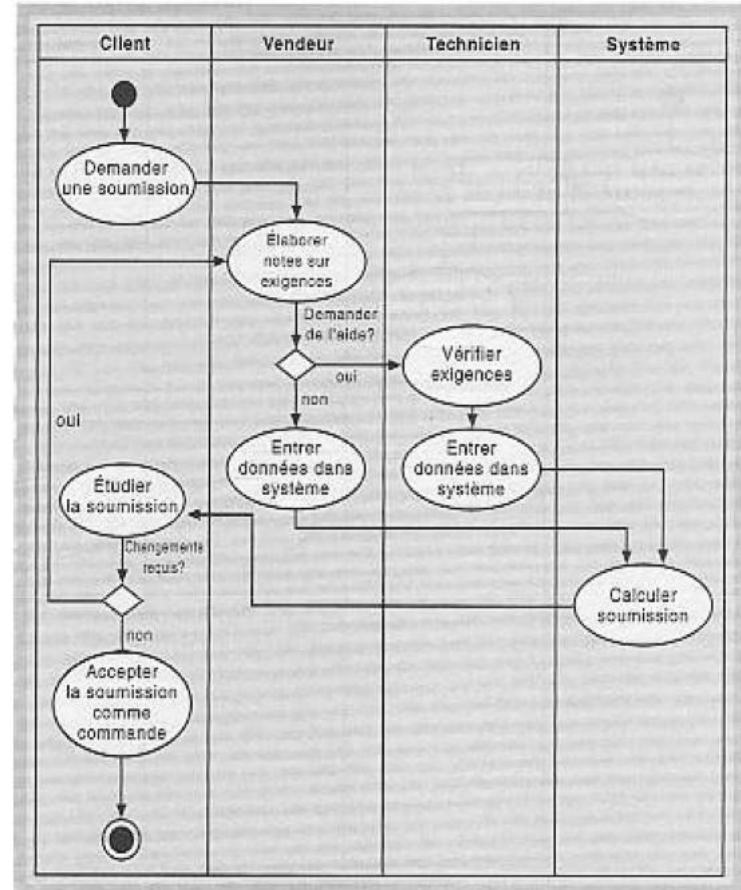
Questions ouvertes subjectives (explicatives)

Collecte des informations

- Observation
 - passive (on regarde seulement) ou active (on participe aux activités)
 - Documenter l'observation (ex: diagramme de flux des travaux)
- Étude des documents et des systèmes logiciels existants
 - Documents d'entreprise: formulaires, procédures de travail, descriptions d'emploi, plans, correspondance, comptes, minutes des réunion, etc.
 - Formulaires et rapports: rapports techniques, saisie d'écran, manuels utilisateurs, modèles d'analyse et de conception, etc.
 - journaux d'affaires, livres de gestion
- Étude des solutions (déjà existantes) des fournisseurs

Collecte des informations

Diagramme
d'activités
représentant
le flux des
travaux



Collecte des informations

- Méthodes actuelles
 - Prototypage
 - Maquette démonstrative, première étude de faisabilité
 - Identification des besoins conflictuels, omis ou mal saisis
 - Prototype jetable:
 - Attention portée sur les besoins les moins bien compris
 - Prototype évolutif
 - Attention portée sur les besoins les mieux compris
 - Développement conjoint d'applications (joint application development - JAD)
 - Série d'ateliers (workshops) ou réunions de travail auxquelles participent clients et développeurs
 - Accélère les communications / feedback rapide
 - Réduit les risques qu'un des participants se manifeste avec de nouvelles exigences

Validation et négociation

- Les besoins répondent-ils aux exigences du client ?
- Réviser la liste des besoins
 - Sont-ils cohérents? Clairs? Intelligibles? ...
 - Élimination des besoins non pertinents ou irréalistes
 - Bien définir les frontières du système
 - Identifier les besoins qui ne répondent pas aux objectifs du système, qui sont hors plan, etc.
 - Faire la liste des besoins exclus pour cause de
 - trop grande difficulté de réalisation
 - mise en œuvre par matériel (hardware)
 - technologie existante inadéquate
 - etc.
- Tout compromis doit être négocié avec le client

Validation et négociation

- Classer les besoins selon leur priorité et évaluer le risque associé à chacun
 - Quels sont les besoins susceptibles de causer des problèmes pendant le développement ?
 - risques techniques
 - risques de performance
 - risques politiques/légaux/éthiques
 - risques de volatilité (besoins modifiés au cours du développement)
 - etc.

Priorisation des besoins

- Les besoins sont habituellement enregistrés sous forme de liste priorisée (backlog)
 - Les développeurs choisissent le prochain changement à effectuer en fonction de cette liste
 - La liste peut être le résultat d'une phase d'analyse dédiée (modèle en cascade) ou élaborée progressivement (modèles agiles)
- Plusieurs critères de priorisation
 - Sévérité
 - Valeur
 - Risque
 - Besoins

Priorisation par sévérité

1. Erreur fatale
2. L'application est sévèrement compromise
 - Impossible de contourner l'erreur
3. Certaines fonctionnalités sont compromises
 - Possible de contourner l'erreur
4. Problème mineur
 - N'implique pas de fonctionnalité critique

Priorisation par valeur d'affaires

1. Fonctionnalité essentielle
 - L'application est inutile
2. Fonctionnalité nécessaire
 - Les utilisateurs comptent sur cette fonctionnalité
3. Fonctionnalité importante
 - L'application est utile même sans cette fonctionnalité
4. Amélioration (mineure)



Priorisation par risque

1. Risque sévère

- Le projet risque d'échouer si non-résolu

2. Risque important

- Ne peut être ignoré

3. Risque éloigné

- Moins sévère, mais requiert l'attention de l'équipe

4. Obstacle mineur



Priorisation par besoins

1. Besoin clé
 - Si implémenté trop tard, la majorité du code devra être réécrit
2. Besoin important
 - Si repoussé, peut forcer à réécrire beaucoup de code
3. Réécriture non négligeable
4. Réécriture mineure / limitée



Gestion des besoins

- Lorsqu'une exigence est changée, comment facilement retracer les documents, modèles et bout de code à modifier?
- Modifications facilitée par l'utilisation d'un outil de gestion de configuration (ex: StarTeam)
 - Permet de tracer
 - Les besoins qui définissent ce que le système doit faire
 - Les modules de conception générés à partir des besoins
 - Le code qui implémente la conception
 - Les tests qui vérifient les fonctionnalités du système

Gestion des besoins

- Identification et classification des besoins dans le cahier des charges
 - identificateur unique, numérotation séquentielle
- Hiérarchisation des besoins
 - Un besoin peut se composer d'un ou plusieurs sous-besoins
 - On peut construire d'abord un modèle abstrait ne considérant pas les sous-besoins plus spécifiques
 - Ex:
 - B1. Le programme doit planifier les activités de l'ascenseur de façon efficace et raisonnable.
 - B1.1 Si l'ascenseur ne contient pas de passager, il devrait demeurer au rez-de-chaussée en attendant la prochaine requête.
 - B1.2 L'ascenseur ne devrait pas modifier le sens de son déplacement s'il contient des passagers qui n'ont pas encore atteint leur destination.

Utilisateurs du cahier des charges

System
Customers

Specify the requirements and read them to check that they meet their needs. Customers specify changes to the requirements.

Managers

Use the requirements document to plan a bid for the system and to plan the system development process.

System
Engineers

Use the requirements to understand what system is to be developed.

System
Test Engineers

Use the requirements to develop validation tests for the system.

System
Maintenance
Engineers

Use the requirements to understand the system and the relationships between its parts.

- Extrait de Ian Sommerville, Page 92

Cahier des charges

1. Description générale du projet

- Intention et portée du projet
- Contexte d'entreprise (planification stratégique)
- Parties prenantes
- Idées de solution
- Plan du document

2. Services du système

- Portée du système (diagramme de contexte)
- Besoins fonctionnels
- Besoins des données (attributs, interrelations)

3. Contraintes du système

- Contraintes d'interface
- Contraintes de performance
- Contraintes de sécurité
- Contraintes opérationnelles
- Contraintes politiques et légales

4. Éléments du projet

- Problèmes ouverts
- Planning préliminaire
- Budget préliminaire

5. Appendices

- Glossaire
- Documents et formulaires d'entreprise
- Références bibliographiques

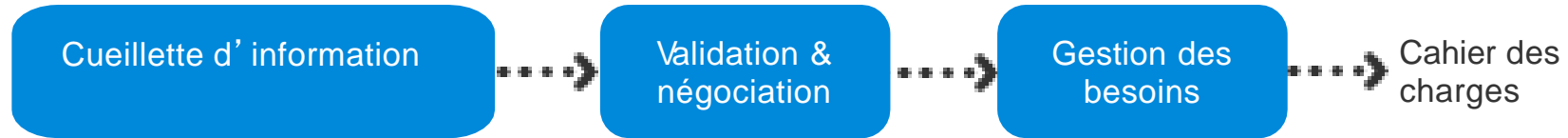
Analyse et Spécifications des besoins:

Spécification et Modélisation des besoins

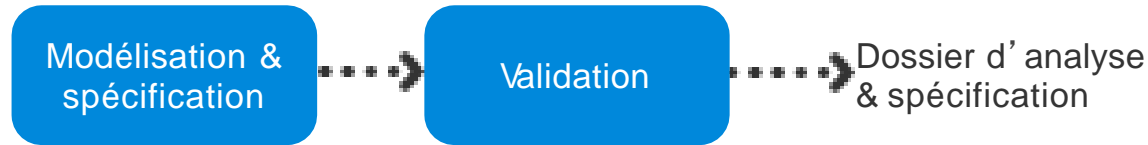


Processus d'analyse des besoins

A. Expression des besoins



B. Spécification et modélisation des besoins



Recueillir l'information.
Définir les caractéristiques du système
Bâtir des prototypes pour la découverte

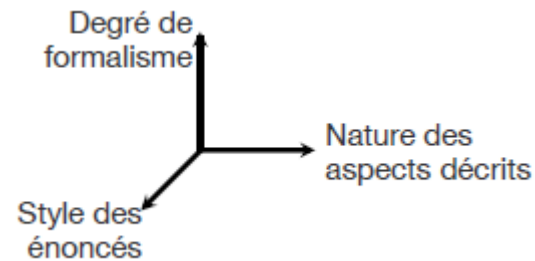
Prioriser les caractéristiques.
Produire et évaluer des solutions de rechange.
Examiner les recommandations

A // B ou A → B

Style de spécification

- Degré de formalisme

- Spécifications informelles
 - langue naturelle, croquis, etc.
- Spécifications semi-formelles
 - Notation graphique dont la sémantique n'est pas formellement définie (ex: UML)
- Spécifications formelles
 - Spécifications algébriques / logiques, réseaux de Petri, langages de programmation, etc.



Style de spécification

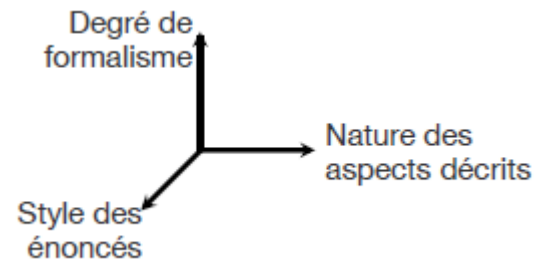
- Style des énoncés

- Spécifications opérationnelles

- ex: soit a un tableau de n éléments. Le résultat de la fonction trier appliquée à a est
 - le premier élément de b est le plus petit de a
 - Le second élément de b est le plus petit du tableau de $n-1$ éléments obtenu
 - Et ainsi de suite jusqu'à ce que les n éléments aient été retirés

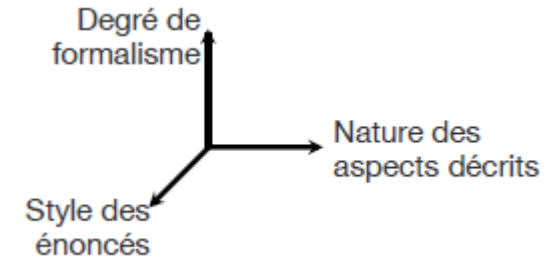
- Spécifications descriptives

- ex: Le résultat de la fonction trier appliquée à a est un tableau b qui est une permutation de a et qui est ordonné



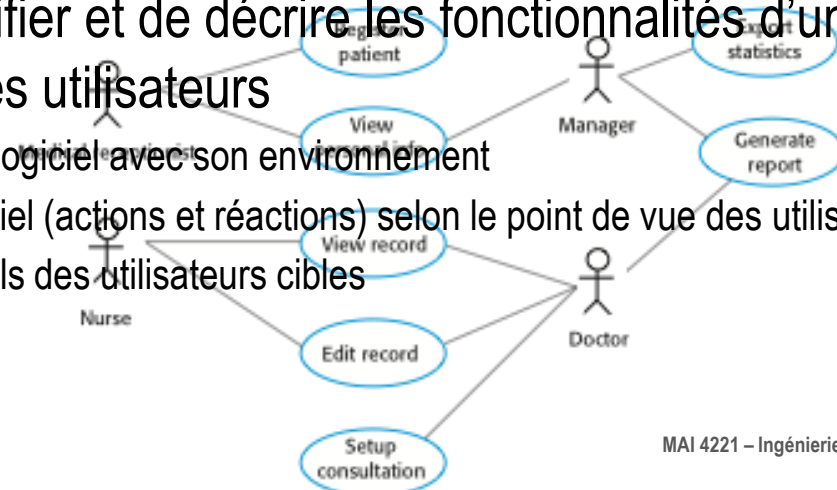
Style de spécification

- Nature des aspects décrits
 - Aspects statiques
 - Ne changent pas
 - ex: format des données, propriétés des fonctions
 - Aspects dynamiques
 - Changent au cours de l'exécution
 - ex: états, réactions aux stimuli



Utilisation du diagramme de cas d'utilisation - UML

- Façon d'organiser ou de traduire les besoins
- Description des scénarios d'utilisation du logiciel
- Identification des services (cas d'utilisation) offerts par le système
- Identification des acteurs participant à chacun des cas d'utilisation
- Description détaillée des scénarios
- Technique permettant d'identifier et de décrire les fonctionnalités d'un logiciel qui sont significatives pour ses utilisateurs
 - Permet de décrire les interactions du logiciel avec son environnement
 - Expression du comportement du logiciel (actions et réactions) selon le point de vue des utilisateurs
 - Détermination des besoins fonctionnels des utilisateurs cibles



Analyse et Spécifications des besoins: **A Retenir...**



A retenir...

- Les besoins pour un système logiciel définissent ce que le système doit faire et définissent des contraintes pour son fonctionnement et sa mise en œuvre.
- Les besoins fonctionnelles sont des déclarations des services que le système doit fournir ou des descriptions de la manière dont certains calculs doivent être effectués.
- Les besoins non fonctionnelles limitent souvent le système en cours de développement et le processus de développement utilisé. Celles-ci peuvent être des besoins de produit, des besoins organisationnelles ou des besoins externes. Ils ont souvent trait aux propriétés émergentes du système et s'appliquent donc à l'ensemble du système.

A retenir...

- Le document relatif aux besoins logiciels est un énoncé convenu des exigences système. Il doit être organisé de manière à ce que les clients du système et les développeurs de logiciels puissent l'utiliser.
- Le processus d'ingénierie des besoins comprend une étude de faisabilité, la détermination et l'analyse des besoins, la spécification des besoins, la validation des besoins et la gestion des besoins.
- L'élicitation et l'analyse des besoins est un processus itératif qui peut être représenté comme une spirale d'activités: découverte des besoins, classification et organisation des besoins, négociation des besoins et documentation des besoins.

A retenir...

- La validation des besoins est le processus de vérification des exigences de validité, de cohérence, d'exhaustivité, de réalisme et de vérifiabilité.
- Les changements commerciaux, organisationnels et techniques entraînent inévitablement des modifications des besoins relatives à un système logiciel. La gestion des besoins est le processus de gestion et de contrôle de ces changements.

