

Chapitre 2 - Recueil des besoins.

BUT 2 - R3.03 Analyse

Halim Djerroud



révision : 0.1

Plan du chapitre

Du cahier des charges au recueil des besoins

- 1 Cahier des charges
- 2 Recueil des exigences
- 3 Modélisation des processus métiers
- 4 Cas d'utilisation (introduction au chapitre suivant)

Cahier des charges

Cahier des charges

- 1 Définition et rôle
- 2 Qualité d'un cahier des charges
- 3 Structure type
- 4 Lecture et interprétation

Définition et rôle du cahier des charges

- **Première étape** de l'expression du besoin dans un projet
- **Rédigé par le client**, en collaboration avec le responsable du développement
- **Décrit les attentes et objectifs** : ce que le système doit faire, dans quel contexte
- Sert de **référence commune** et de **contrat** entre les acteurs du projet
- Document **technique**, centré sur les besoins fonctionnels (pas de justification économique)

Pourquoi un cahier des charges ?

- Document fondateur d'un projet informatique
- Permet d'aligner **client, utilisateurs et développeurs**
- Évite les malentendus et les interprétations différentes
- Sert de référence tout au long du cycle de vie
- Point de départ du recueil des exigences

Qualité d'un cahier des charges

- **Niveau de généralité adapté** : ni trop vague, ni trop détaillé
- **Clarté et absence d'ambiguïté** : termes précis, compréhensibles par tous
- **Critères de validation** : chaque exigence doit être vérifiable
- **Souplesse** : permettre l'évolution des besoins en cours de projet
- **Concision et réalisme** : document synthétique, atteignable dans les contraintes

Structure type

- ➊ Introduction : Information générale sur le document, les objectifs du document, sa structure et les documents référencés.
- ➋ Énoncé : Description détaillée du problème à résoudre, le contexte, les objectifs du projet. Si besoin, on fait une présentation de l'existant. Définition des objectifs que doit atteindre la solution
- ➌ Pré-requis : Connaissances requises, ressources matérielles et logicielles, compétences nécessaires
- ➍ Priorités : es priorités éventuelles du développements si elles ont été fixées avec l'accord du client.

Lecture et interprétation du cahier des charges

- **Identifier les éléments du problème** : objets, acteurs et actions associés
- **Repérer les exigences fonctionnelles** : ce que le système doit réaliser
- **Analyser le contexte** : interfaces externes, formats de données, règles métier
- **Compléter et affiner** : enrichir avec le recueil d'exigences auprès du client et des utilisateurs

Identifier les objets et acteurs

- **Objets** : éléments manipulés par le système
 - Exemples : commande, produit, facture, utilisateur
- **Acteurs** : entités qui interagissent avec le système
 - Exemples : client, employé, administrateur, système externe
- **But** : lister tous les éléments nécessaires à la compréhension du domaine

Identifier les actions

- Définir les **actions principales** des acteurs sur les objets
 - Exemple : un client *pass*e une commande
 - Exemple : un administrateur *valid*e une facture
- Les actions révèlent les **exigences fonctionnelles** attendues
- Chaque action doit être **claire, précise et non ambiguë**
- Ces éléments servent de base à la rédaction des **cas d'utilisation**

Exemple : Objets, acteurs et actions

Objet	Acteur	Action
Commande	Client	Passer une commande
Facture	Administrateur	Valider une facture
Produit	Employé	Mettre à jour le stock
Compte	Utilisateur	Consulter son solde

Ce tableau aide à relier les éléments du domaine aux exigences fonctionnelles.

Recueil des exigences

Recueil des exigences

- ❶ Objectifs et portée du recueil
- ❷ Glossaire et terminologie
- ❸ Cas d'utilisation (acteurs, objectifs, scénarios)
- ❹ Autres exigences et contraintes

Recueil des exigences

- Chaque organisation recueille les exigences correspondant à ses besoins
- Il existe des **standards** pour décrire ces exigences
- Pour guider ce travail, il faut se poser un certain nombre de **questions clés**
- Alistair Cockburn (2009) propose une adaptation du plan de Suzanne Robertson (1999)
- Plan structuré en **6 chapitres**
- Ce plan, très complet, doit être **adapté et simplifié** selon les projets

Plan des exigences (Cockburn, 2009)

❶ Objectif et portée

- Objectifs généraux
- Intervenants
- Limites du système

❷ Terminologie employée

- Glossaire partagé
- Définitions métier

❸ Cas d'utilisation

- Acteurs
- Objectifs
- Scénarios

❹ Technologie employée

- Exigences techniques
- Interfaces

❺ Autres exigences

- Processus, règles métier
- Performances, sécurité
- Maintenance, portabilité

❻ Facteurs humains

- Ressources humaines
- Aspects juridiques et politiques
- Conséquences humaines

Chapitre 1 – Objectif et portée

- **Objectifs généraux** : pourquoi le projet existe ? quels résultats attendus ?
- **Portée du système** :
 - Ce qui est inclus (fonctionnalités à développer)
 - Ce qui est exclu (hors périmètre du projet)
- **Intervenants concernés** :
 - Clients et utilisateurs finaux
 - Développeurs et responsables techniques
 - Partenaires ou systèmes externes

Chapitre 2 – Terminologie employée

- Définir un **glossaire partagé** pour éviter les malentendus
- Uniformiser les termes **métier** et **techniques**
- Facilite la **communication** entre toutes les parties prenantes
- Exemple :
 - « Client » = personne qui passe une commande
 - « Utilisateur » = personne qui utilise le logiciel

Chapitre 3 – Cas d'utilisation

- Identifier les **acteurs principaux** et leurs objectifs
- **Cas d'utilisation métier** :
 - Décrivent les processus et objectifs de l'organisation
 - Ex. : « Le client souhaite acheter un produit »
- **Cas d'utilisation système** :
 - Décrivent comment le logiciel soutient ces objectifs
 - Ex. : « Le système enregistre la commande et envoie une confirmation »
- Les cas d'utilisation constituent la **base des exigences fonctionnelles**

Chapitre 4 – Technologie employée

- **Exigences technologiques** : choix imposés de matériel, de logiciels ou de standards
 - Ex. : base de données SQL, système embarqué, langage imposé
- **Interfaces avec d'autres systèmes** : préciser comment le système doit communiquer avec l'existant
 - Ex. : API externe, compatibilité avec ERP ou CRM
- **Contraintes matérielles et logicielles** : limitations à respecter
 - Ex. : fonctionner sur terminaux mobiles, supporter Windows et Linux

Chapitre 5 – Autres exigences

- **Processus de développement** : méthodes, valeurs clés, visibilité, stratégie (acheter ou construire), gestion des tests
- **Règles métier** : contraintes propres à l'organisation (ex. validation hiérarchique)
- **Performances attendues** : rapidité, disponibilité, temps de réponse
- **Qualité logicielle** : sécurité, documentation, ergonomie et utilisabilité
- **Maintenance et portabilité** : mises à jour, compatibilité multi-plateformes
- **Questions ouvertes** : points reportés ou dépendances externes

Chapitre 6 – Facteurs humains et organisationnels

- **Ressources humaines** : rôles et responsabilités des personnes impliquées (ex. administrateurs, support utilisateur)
- **Exigences juridiques et politiques** : conformité avec la loi, normes ou règles internes (ex. RGPD, réglementation sectorielle)
- **Conséquences humaines** : besoins en formation, impacts sur l'organisation du travail
- **Hypothèses et dépendances** : contraintes externes influençant le projet (ex. dépendance à un fournisseur, disponibilité du matériel)

Modéliser les processus métiers

Pourquoi modéliser les processus métiers ?

- 1 Comprendre le **fonctionnement de l'organisation**
- 2 Identifier les **exigences organisationnelles**
- 3 Visualiser la **structure et la dynamique** du travail
- 4 Support pour la communication entre client et développeur
- 5 Préparer la rédaction des **cas d'utilisation**

Outils de modélisation : UML

- Utilisation des **diagrammes d'activités UML**
- Permettent de représenter :
 - Le déroulement d'un processus
 - Les décisions et alternatives
 - Les enchaînements d'actions
- Peuvent aussi spécifier le **déroulement d'un cas d'utilisation**

Analyse des exigences organisationnelles

- Identifier les **rôles et responsabilités** dans l'organisation
- Déterminer les **flux d'informations** entre acteurs
- Repérer les **contraintes métier** à respecter
- Fournir une base solide pour les **exigences comportementales**

Exemple de processus métier

- Contexte : gestion d'une commande en ligne
- Étapes :
 - ① Le client sélectionne des produits
 - ② Le système génère un panier
 - ③ Le client valide et paie
 - ④ Le système enregistre et envoie une confirmation
- Représentation possible : **diagramme d'activités UML**