

CAHIER DES CHARGES

2

AGL – Chapitre

Mme. Lilia SFAXI
Mme. Abir Gallas

L2ARS/SIL – 2011/2012

Définition du Cahier des Charges

- Le Cahier des Charges (CDC) d'un projet est un document par lequel on exprime son besoin pour le projet.
- Ce besoin doit être formulé en termes de fonctions que le futur utilisateur aura à accomplir, ou que le système devra accomplir pour lui.

Définition du Cahier des Charges (2)

- Le CDC permet en outre :
 - ✓ de provoquer chez le concepteur / réalisateur (prestataire) la conception et la réalisation du produit le plus efficient,
 - ✓ de faciliter le dépouillement des propositions des prestataires,
 - ✓ de favoriser le dialogue entre les partenaires.

Définition AFNOR

Document par lequel le demandeur exprime son besoin (ou celui qu'il est chargé de traduire) en terme de fonctions de services et de contraintes. Pour chacune d'elles sont définis des critères d'appréciation et leurs niveaux. Chacun de ces niveaux doit être assorti d'une flexibilité.

Produire un CDC : Méthodologie

- Le CDC doit être rédigé indépendamment des concepts de solutions envisageables afin de laisser le plus grand éventail de concepts de solutions possibles.
- Le CDC doit permettre au maximum l'expression du besoin dans les termes des différents utilisateurs selon les phases de l'état vivant du produit.
- Le CDC relate les besoins exactes des utilisateurs. Pour ce faire, des entretiens sont menés et un groupe de travail est constitué.

Produire un CDC : Méthodologie (2)

- Le Cahier des Charges Fonctionnel est la conclusion des travaux d'analyse de la valeur et d'analyse fonctionnelle qui symbolisent la démarche d'expression du besoin :
 - ✓ Orienter l'étude : Du général au spécifique.
- Premiers points de la démarche :
 - ✓ regarder le projet d'un œil extérieur
 - ✓ prendre du recul
 - ✓ se poser les bonnes questions :
 - Rechercher l'information
 - Traduire le besoin en fonctions
 - Formaliser les travaux
 - Contrôler le CDC Besoin
 - Valider le CDC Besoin

Rechercher l'information

- La recherche de l'information doit être canalisée et formalisée.
- C'est un processus constant tout au long du projet qui doit être mené rigoureusement dès le début du projet afin d'appréhender plus précisément les caractéristiques essentielles du besoin.
- Un excellent moyen de chercher l'information la plus pertinente et de la vérifier en même temps est de constituer un groupe de travail.

Traduire le besoin en fonctions

- Le passage du besoin en fonction s'effectue au travers de l'analyse fonctionnelle qui recense, caractérise, ordonne, hiérarchise et valorise les fonctions.

Formaliser les travaux

- Cette formalisation consiste à développer le Cahier des Charges. Il reprendra les conclusions de l'analyse fonctionnelle.

Contrôler le CDC besoin

- Le contrôle du document est très important. En effet, on remarque que cette étape n'est généralement pas effectuée de façon optimale alors qu'elle est un frein aux dysfonctionnements qui peuvent apparaître beaucoup plus tard dans le projet.

Valider le CDC besoin

- Il s'agit de s'assurer que le passage du besoin exprimé au besoin fonctionnel est conforme aux objectifs visés. C'est un travail qui peut s'avérer fastidieux et risqué si le volume d'information est important. L'objectif est donc ici de rendre efficace la validation en réduisant son domaine d'action et tout en conservant sa représentativité.

Exemple de CDC selon IEEE std 830

I- Introduction

II- Contexte de la réalisation

1. Objectifs
2. Hypothèses
3. Bases méthodologiques

III- Description générale

1. Environnement du projet
2. Fonctions générales du système
3. Caractéristiques des utilisateurs
4. Configuration du système
5. Contraintes générales du développement, d'exploitation et de maintenance
 - ✓ Contraintes de développement
 - ✓ Contraintes d'exploitation
 - ✓ Maintenance et évolution du système

Exemple de CDC selon IEEE std 830 (2)

IV- Description des interfaces externes du logiciel

1. Interface matériel / logiciel
2. Interface homme / machine
3. Interface logiciel / logiciel

V- Description des objets

1. Définition des objets
 - ✓ Identification de l'objet i
 - ✓ Contraintes sur l'objet i

VI- Description des fonctions

1. Définitions des fonctions
 - ✓ Identification de la fonction i
 - ✓ Description de la fonction i
 - ✓ Contraintes opérationnelles sur la fonction i

Exemple de CDC selon IEEE std 830 (3)

2. Conditions particulières de fonctionnement

- 2.1. Performances
- 2.2. Capacités
- 2.3. Modes de fonctionnement
- 2.4. Contrôlabilité
- 2.5. Sûreté
- 2.6. Intégrité
- 2.7. Conformité aux standards
- 2.8. Facteurs de qualité

VII- Justification des choix effectués

VIII- Glossaires

IX- Références

- 1. Annexes
- 2. Index

Exemple simple: Gestion d'une bibliothèque

Fonctionnalités

- ✓ Il s'agit d'un outil d'aide à la gestion de bibliothèque.
- ✓ Une bibliothèque prête des livres et des magazines à des emprunteurs.
- ✓ Les livres et les magazines sont répertoriés dans le système.
- ✓ Les emprunteurs sont répertoriés dans le système.
- ✓ Une bibliothèque s'occupe de l'achat de nouveaux titres.
- ✓ Les titres populaires sont achetés en plusieurs exemplaires.
- ✓ Les vieux livres ou magazines sont retirés lorsqu'ils sont trop anciens.
- ✓ Les vieux livres ou magazines sont retirés lorsqu'ils sont en mauvais état.
- ✓ Le bibliothécaire est un employé de la bibliothèque.

Exemple simple: Gestion d'une bibliothèque (2)

- Le bibliothécaire communique avec les emprunteurs.
- L'outil assiste le bibliothécaire dans sa tâche.
- Un emprunteur peut réserver un livre ou un magazine qui n'est pas disponible (déjà prêté ou non encore répertorié).
- Lorsqu'un livre ou un magazine devient disponible (rendu ou acheté), l'emprunteur qui l'a réservé est averti.
- La réservation est annulée lorsque le livre ou le magazine est prêté.
- Une réservation peut être annulée à tout moment.
- Les création, mise à jour et destruction d'informations relatives aux titres, emprunteurs, prêts et réservations doivent être aisées.

Exemple simple: Gestion d'une bibliothèque

- Constraintes non fonctionnelles
 - ✓ L'application doit tourner dans tout environnement Unix ou Windows.
 - ✓ L'application doit avoir une IHM agréable.
 - ✓ L'application doit pouvoir être étendue à d'autres fonctionnalités.
- Limitations
 - ✓ L'application ne gère pas l'envoi du message d'avertissement aux emprunteurs lorsque le livre ou le magazine qu'ils ont réservé devient disponible.
 - ✓ L'application ne gère pas les retards à la restitution.

Exercice : Gestion de projets

- Une société de développement logiciel décide d'implémenter son propre outil de gestion de projet. Elle a dégagé les entités suivantes :
- Un projet est caractérisé par son identifiant, son nom, une description, une date de début, une date de fin.
- Un projet passe par plusieurs phases. Chaque phase est caractérisée d'un identifiant, un nom, une description, une date de début, une date de fin et réalisée par une équipe de personnes dont l'un est responsable (il existe un seul responsable pour une phase). Une phase doit générer un rapport.
- Chaque document est caractérisé par son identifiant, son nom, une description, sa date de validation, son état (valide, non valide, en attente).
- Une personne est caractérisée par son identifiant, son nom, son prénom, son âge. A un instant il participe à une seule phase.

Exercice : Gestion de projets (2)

- Donner la description textuelle de ce logiciel en se basant sur le modèle suivant :

Modèle de la description textuelle

- Résumé :
Titre, But, Résumé, Dates, Version, Responsables, Acteurs,...
- Préconditions :
Ce qui doit être vérifié avant que le C.U. ne commence
- Enchaînements :
 - Evènements de déclenchement
 - Séquence nominale/C.U. référencés
 - Séquences Exceptionnelles/Exceptions
- Postconditions
ce qui est vrai après que le C.U. se soit déroulé

Exercice : Gestion de projets (3)

- Besoins d'IHM
Expression de contraintes liées à l'interface
- Contraintes non fonctionnelles
 - Fréquences
 - Volumétries
 - QoS: Disponibilité, fiabilité, Performances
 - Concurrence

Exercice : Gestion de projets (Correction)

1. Fiche descriptive

a. Résumé

- Titre : Logiciel de gestion de projet
- But : Automatisation de la gestion de projet
- Résumé : Le logiciel va permettre une gestion complète, efficace et rapide d'un projet
- Date : 15-02-2010
- Version : 1.0
- Responsable : le chef de projet Mr X
- Acteurs : 1 ingénieur conception, 3 ingénieurs développement, 2 ingénieurs intégration et 2 ingénieurs validation

Exercice : Gestion de projets (Correction)

b. Pré conditions

- Il faut avoir 10 machines en bon état avec un OS linux et tout les logiciels de conception, développement, intégration et validation Une phase ne doit avoir qu'un seul responsable.
- Un acteur ne peut intervenir qu'à une seule phase à la fois.

c. Enchaînement

- Evènement de déclenchement : l'arrivée d'un nouveau projet
- Séquences nominales : le logiciel doit gérer toutes les phases d'un cycle de vie d'un logiciel : conception, développement, intégration, tests et validation, documentation et la maintenance
- Séquences exceptionnelles : Si un projet ne nécessite pas une des phases ou le client demande de la sauter, le logiciel devra prendre en considération ce changement

Exercice : Gestion de projets (Correction)

d. Post conditions

- Chaque phase doit générer un document

e. Besoin d'IHM

- Ce logiciel doit avoir une interface facile à gérer

f. Contraintes non fonctionnelles : ce logiciel doit être portable, fiable